REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ORGANIZATION.

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
07/02/2012	Final Report	December 1,2008-November 30, 2011

4. TITLE AND SUBTITLE
GEOMETRIC METHODS FOR CONTROLLED ACTIVE VISION

5a. CONTRACT NUMBER
FA9550-09-1-0172

5b. GRANT NUMBER
FA9550-09-1-0172

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Allen Tannenbaum

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Georgia Tech, Atlanta, Georgia 30332

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AFOSR
875 N. Randolph St.
Suite 325
Arlington, VA 22203

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
AFRL-OSR-VA-TR-2012-0383

12. DISTRIBUTION/AVAILABILITY STATEMENT
A

13. SUPPLEMENTARY NOTES

14. ABSTRACT

In this just completed research program, we developed several new directions for our work in controlled active vision. We have developed a general framework for geometric observer-like structures based on non-parametric implicit (level set) curve descriptions of dynamically varying shapes. Special emphasis was given to the geometric nature of the dynamical system as well as the key issue of robustness. In particular, we formulated an approach to the problem of information transport and filtering from a measurement curve to an estimated curve. In this framework, we may naturally incorporate several different tools such as particle filtering and segmentation methods.

15. SUBJECT TERMS
controlled active vision, visual tracking, information theory, curvature based flows

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER *(Include area code)*
U	U	U	UU		

Reset

FINAL REPORT FOR GEOMETRIC METHODS FOR CONTROLLED ACTIVE VISION
Grant/Contract Number: FA9550-09-1-0172
PI: Allen Tannenbaum

Abstract

In this just completed research program, we developed several new directions for our work in controlled active vision. We have developed a general framework for geometric observer-like structures based on non-parametric implicit (level set) curve descriptions of dynamically varying shapes. Special emphasis was given to the geometric nature of the dynamical system as well as the key issue of robustness. In particular, we formulated an approach to the problem of information transport and filtering from a measurement curve to an estimated curve.

We also formulated a methodology to assess measurement reliability that allows for the selection of a local observer-gain. We should note that the dynamical nature of the evolving curves described implicitly allows for the observation of objects changing topology (i.e., objects breaking and merging during propagation) for which shape priors can be naturally incorporated. The proposed observer structure is continuous/discrete, with continuous-time system dynamics and discrete-time measurements. Its state space consists of an estimated curve position augmented by additional states (e.g., velocities) associated with every point on the estimated curve. Multiple simulation models were proposed for state prediction. Measurements are performed through segmentation algorithms and optical flow computations.

In this framework, we may naturally incorporate several different tools such as particle filtering as well as various segmentation procedures. Accordingly, we described a novel information-based segmentation algorithm, which because of its local/global nature seems ideal for tracking and was combined with particle filtering for robust tracking in our geometric observer approach.

Papers of Allen Tannenbaum and Collaborators under AFOSR Support Since 2008

1. "Fast approximation of smooth functions from samples of partial derivatives with applications to phase unwrapping" (with O. Michailovich), *Signal Processing* **88** (2008), pp. 358-374.

2. "Finsler active contours" (with E. Pichon and J. Melonakos), *IEEE PAMI* **30** (2008), pp. 412-423.

3. "Geometric observers for dynamically evolving curves" (with P. Vela and M. Niethammer), *IEEE PAMI* **30** (2008), pp. 1093-1108.

4. "Knowledge-based segmentation for tracking through deep turbulence" (with P. Vela, M. Niethammer, G. Pryor, R. Butts, and D. Washburn), *IEEE Trans. Control Technology* **16** (2008), pp. 469-475.

5. "A framework for image segmentation using shape models and kernel space shape priors" (with S. Dambreville and Y. Rathi), *IEEE PAMI* **30** (2008), pp. 1385-1399.

6. "Dynamic denoising of tracking sequences" (with O. Michailovich and Y. Rathi), *IEEE Trans. Image Processing* **17** (2008), pp. 847-857.

7. "Segmentation of tracking sequences using dynamically updated adaptive learning" (with O. Michailovich), *IEEE Trans. on Image Processing* **17** (2008), pp. 2403-2413.

8. "Localizing region-based active contours" (with S. Lankton), *IEEE Trans. Image Processing* **17** (2008), pp. 2029-2039.

9. "Choice of in vivo versus idealized velocity boundary conditions influences the flow field in subject-specific models of the human carotid bifurcation" (with A. Wake, D. Giddens, J. Oshinsky), *Journal of Biomedical Engineering* **131** (2009).

10. "Real-time visual tracking using geometric active contours and particle filters" (with Jincheol Ha and E. Johnson), *Journal of American Institute of Aeronautics and Astronautics* **5**(10) (2008), pp. 361-379.

11. "Vision-based range regulation of a leader-follower formation" (with P. Vela, A. Betser, J. Malcolm), *IEEE Trans. Control Technology* **17** (2009), pp. 442-449.

12. "Near tubular fiber bundle segmentation for diffusion weighted imaging: segmentation through frame reorientation" (with M. Niethammer and C. Zach), *NeuroImage* **45** (2009), pp. 123-132.

13. "An efficient numerical method for the solution of the L^2 optimal mass transfer problem" (with E. Haber and T. Rehmen), *SIAM Journal on Scientific Computation* **32** (2010), pp. 197-211.

14. "3D nonrigid registration via optimal mass transport on the GPU" (with E. Haber, T. Rehman, G. Pryor, and J. Melonakos), *MedIA* **13** (2009), pp. 931-940.

15. "Deform PF-MT (Particle Filter with Mode Tracker) for tracking nonaffine contour deformations," (with N. Vaswani, Y. Rathi, and A. Yezzi), *IEEE Trans. Image Processing* **19** (2010), pp. 841-857.

16. "A geometric approach to joint 2D region-based segmentation and 3D pose estimation using a 3D shape prior" (with S. Dambreville, R. Sandhu, A. Yezzi), *SIAM Imaging Science* **3** (2010), pp. 110-132.

17. "Targeting of PbSe-Fe2O3 nanoplatforms by external magnetic field under viscous flow conditions," (with Lioz Etgar, Arie Nakhmani, Efrat Lifshitz, and Rina Tannenbaum), *Sensor Lett.* **8** (2010), pp. 383-386.

18. "Trajectory control of PbSe - $\gamma - Fe_2O_3$ nanoplatforms under viscous flow and an external magnetic field" (with L. Etgar, A. Nakhmani, E. Lifshitz, R. Tannenbaum), *Nanotechnology* **21** (2010), pp. 1-9.

19. "Texture mapping via optimal mass transport" (with A. Dominitz), *IEEE TVCG* **16** (2010), pp. 419-433.

20. "Imaging of meningioma progression by matrix-assisted laser desorption/ionization time-of-flight mass spectrometry" (with N. Agar, J. Malcolm, V. Mohan, H. Yang, M. Johnson, J. Agar, P. Black), *Analytical Chemistry* **82** (2010), pp. 2621-2625.

21. "Affine registration of label maps in label space" (with Y. Rathi, J. Malcolm, S. Bouix, M. E. Shenton), *Journal of Computing* **2**:4 (2010), pp. 1–11.

22. "Relevance vector machine learning for neonate pain intensity assessment using digital imaging" (with B. Gholami and W. Haddad), *IEEE Trans. Biomedical Engineering* **57** (2010), pp. 1457-1466.

23. "Point set registration via particle filtering and stochastic dynamics," (with R. Sandhu and S. Dambreville), *IEEE PAMI* **32** (2010), pp. 1459-1473.

24. "A non-rigid kernel based framework for 2D/3D pose estimation and 2D image segmentation" (with R. Sandhu and S. Dambreville), *IEEE PAMI* **33**, pp. 1098-1115.

25. "Tubular surface segmentation for extracting anatomical structures from medical imagery" (with V. Mohan and G. Sundaramoorthi), *IEEE Trans. Medical Imaging* **29** (2010), pp. 1945-1958.

26. "A coupled global registration and segmentation framework with application to magnetic resonance prostate imagery" (with Y. Gao and R. Sandhu), *IEEE Trans. Medical Imaging* **29** (2010), pp. 1781-1794.

27. "Non-parametric clustering for studying RNA conformations" (with X. LeFaucheur, E. Hershkovits, R. Tannenbaum), *IEEE Trans. Computational Biology and Bioinformatics* **8** (2011), pp. 1604-1618.

28. "Development of stereotactic mass spectrometry for brain tumor surgery," (with N. Agar, V. Mohan, A. Golby, F. Jolescz), *NeuroSurgery* **68** (2011), pp. 280-290.

29. "Object tracking and target reacquisition on 3D range data with particle filtering and online shape learning" (with J. Lee and S. Lankton), *IEEE Trans. Image Processing* **20** (2011), pp. 2912-2924.

30. "Particle filtering with region-based matching for tracking of partially occluded and scaled targets" (with A. Nakhmani), *SIAM Journal Imaging Science* **4** (2011), pp. 220-242.

31. "Clinical decision support and closed-loop control for cardiopulmonary management and intensive care unit sedation using expert systems," (with B. Gholami, W. Haddad, J. Bailey), to appear in *IEEE Transactions on Information Technology in Biomedicine*, 2012.

32. "3D automatic segmentation of the hippocampus using wavelets with applications to radiotherapy planning" (with Y. Gao, B. Corn, D. Schifter), to appear in *MedIA*, 2012.

33. "Self-crossing detection and location for parametric active contours" (with Arie Nakhmani), to appear in *IEEE Trans. Image Processing*, 2012.

34. "Global optimal diffeomorphic registration for point sets," (with Y. Gao and Y. Rathi), submitted to *MedIA*, 2012.

35. "Interactive multi-object segmentation using local robust statistically driven active contours," (with Y. Gao and R. Kikinis), submitted to *IEEE Trans. Image Processing*, 2012.

36. "Particle filters and occlusion handling for rigid 2D-3D pose tracking," (with J. Lee and R. Sandhu), submitted for publication to *IEEE Trans. Image Processing*, 2012.

37. "Volumetric mapping of genus 0 volumes via mass preservation" (with A. Dominitz and R. Sandhu), submitted for publication to *IEEE Trans. Visualization and Computer Graphics*, 2012.

38. "Multimodal deformable image registration via the Bhattacharyya distance" (with Y. Lou), submitted for publication to *MeDIA*, 2012.

39. "Particle filtering for registration of 2D and 3D point sets" (with R. Sandhu), *IEEE CVPR*, 2008.

40. "Multiple object tracking through graph cuts" (with J. Malcolm), *IWCIA*, 2008.

41. "Thresholding active contours" (with S. Dambreville and A. Yezzi), *IEEE ICIP*, 2008.

42. "Tracking through changes in scale" (with S. Lawton, S. Malcolm, A. Nakhmani), *IEEE ICIP*, 2008.

43. "Label space: A coupled multi-shape representation" (with J. Malcolm and Y. Rathi), *MICCAI*, 2008.

44. "Localized statistics for DW-MRI fiber bundle segmentation" (with S. Lankton and J. Melonakos), *MMBIA*, 2008.

45. "Particle filtering using multiple cross-correlations for tracking occluded objects in cluttered scenes" (with A. Nakhmani), *IEEE CDC*, 2008.

46. "Scale-invariant visual tracking by particle filtering" (with A. Nakhmani), *SPIE*, 2008.

47. "Adaptive Bayesian shrinkage model for spherical wavelet based denoising and compression of hippocampus shapes" (with X. LeFaucheur and B. Vidakovic), *Int. Conference on Medical Image Computing and Computer Assisted Intervention*, 2008.

48. "Tubular surface evolution for segmentation of tubular structures with applications to the cingulum bundle from DW-MRI" (with V. Mohan, G. Sundaramoorthi, M. Niethammer), *MFCA'08*, 2008.

49. "Multimodal registration of white matter brain data via optimal mass transport" (with E. Haber, T. Rehmen, K. Pohl, R. Kikinis), *Computational Biomechanics for Medicine*, 2008.

50. "Robust 3D pose estimation and efficient 2D region-based segmentation from a 3D shape prior" (with R. Sandhu and S. Dambreville), *ECCV*, 2008.

51. "Non-rigid 2D-3D pose estimation and 2D image segmentation" (with R. Sandhu and S. Dambreville), *CVPR*, 2009.

52. "Statistical shape learning for 3D tracking" (with S. Lankton and R. Sandhu), *IEEE CDC*, 2009.

53. "Experience with highly automated unmanned aircraft performing complex missions" (with N. Rooz, E. Johnson *et al.*), *AIAA Guidance, Navigation, and Control Conference*, 2009.

54. "Area stabilized visual closed-loop tracking" (with P. Karasev and P. Vela), *Proceedings ACC*, 2010.

55. "Closed-loop control for intensive care unit sedation using expert systems" (with B. Gholami, J. Bailey, W. Haddad), *Proceedings ACC*, 2010.

56. "Segmentation of the epicardial wall of the left atrium using statistical shape learning and local curve statistics" (with Y. Gao, B. Gholami, R. MacLeod, J. Blauer, W. M. Haddad), *SPIE Med. Imag.*, San Diego, CA, 2010.

57. "An unsupervised learning approach to facial expression recognition using semi-definite programming and generalized principal component analysis" (with B. Gholami and W. M. Haddad, *IS&T/SPIE Elec. Imag.*, San Jose, CA, 2010.

58. "Intraoperative prediction of tumor cell concentration from Mass Spectrometry Imaging" (with V. Mohan, N. Agar, F. Jolesz), *SPIE Medical Imaging*, 2010.

59. "Fire and smoke detection in video with optimal mass transport based optical flow and neural networks" (with Ivan Kolesov, Peter Karasev, Eldad Haber), *ICIP*, 2010.

60. "Range based object tracking and segmentation" (with J. Lee), *ICIP*, 2010.

61. "Coupled segmentation for anatomical structures by combining shape and relational spatial information" (Ivan Kolesov, Vandana Mohan, Gregory Sharp), *MTNS*, 2010.

62. "High resolution analysis via sparsity-inducing techniques: spectral lines in colored noise" (with L. Ning and T. Georgiou), *MTNS*, 2010.

63. "Enhanced tubular shape prior for robust segmentation of multiple fiber bundles from brain DWI" (with Vandana Mohan, Ganesh Sundaramoorthi, Marek Kubicki, Douglas Terry and Allen Tannenbaum), *MTNS*, 2010.

64. "Closed-loop control for cardiopulmonary management and intensive care unit sedation using expert systems" (with B. Gholami, W. Haddad, and J. Bailey), *IEEE CDC*, 2010.

65. "Optimal drug dosing control for intensive care unit sedation using a hybrid deterministic-stochastic pharmacokinetic and pharmacodynamic model" (with B. Gholami, W. Haddad, and J. Bailey), *IEEE CDC*, 2010.

66. "Signals & control aspects of optimal mass transport and the Boltzmann entropy" (with E. Tannenbaum and T. Georgiou), *IEEE CDC*, 2010.

67. "Separation of system dynamics and line spectra via sparse representation" (with L. Ning and T. Georgiou), *IEEE CDC*, 2010.

68. "Delay estimation for wireless LAN control of nonlinear systems" (with P. Karasev and P. Vela), *IEEE CDC*, 2010.

69. "Range based object tracking and segmentation," (with J. Lee and P. Karasev) *IEEE International Conference on Image Processing*, pp. 4641-4644, 2010.

70. "Human body tracking and joint angle estimation from mobile-phone video for clinical analysis," (with J. Lee, P. Karasev and L. Zhu), *IAPR Conference on Machine Vision Applications,* 2011.

71. "Monte Carlo sampling for visual pose tracking," (with J. Lee and R. Sandhu), *IEEE International Conference on Image Processing*, 2011.

72. "Estimation of myocardial volume at risk from CT angiography," (with Liangjia Zhu, Yi Gao, Vandana Mohan, Arthur E. Stillman, Tracy Faber), *SPIE Medical Imaging*, 2011.

73. "Fire and smoke detection in video with optimal mass transport based optical flow and neural networks," (with I. Kolesov and P. Karasev), *IEEE International Conference on Image Processing*, 2011.

74. "Interactive MRI segmentation with controlled active vision," (with P. Karasev, I. Kolesov, K. Chudy), *IEEE CDC*, 2011.

75. "Depth invariant visual servoing," (with P. Karasev, M. Serrano, P. Vela), *IEEE CDC*, 2011.

76. "Human body joints estimation for clinical jump analysis," (with L. Zhu, J. Lee, P. Karasev, I. Kolesov, J. Xerogeanes) *International Conference on Medical Image Computing and Computer Assisted Intervention Workshop: Computational Biomechanics for Medicine VI*, 2011.

77. "Stochastic point set registration," (with I. Kolesov, J. Lee, P. Vela, and A. Tannenbaum), submitted for publication in *ECCV*, 2012.

78. "Complexes of nucleic acids with group I and II cations" (with C. Hsiao, E. Tannenbaum, E. Hershkovitz, G. Perng, S. Howerton , and L Williams), *Nucleic Acid-Metal Ion Interactions* edited by Nicholas Hud, pages 1-38, RSC Publishing, Cambridge, UK, 2008.

79. "On the computation of optimal transport maps using gradient flows and multiresolution analysis" (with A. Dominitz and S. Angenent), *Recent Advances in Learning and Control* edited by V. Blondel, S. Boyd, and H. Kimura, Springer-Verlag, New York, 2008.

80. "Label space: A multi-object representation" (with J. Malcolm and Y. Rathi), in *Combinatorial Image Analysis*, Lecture Notes in Computer Science **4958** (2008), pp. 185-196.

81. "Sparse blind source deconvolution with application to high resolution frequency analysis" (with T. Georgiou), *Three Decades of Progress in Control Sciences*, X. Hu, U. Jonsson, B. Wahlberg, B. Ghosh (Eds.), Springer-Verlag, 2010.

82. "Sparse blind source separation via ℓ_1-norm optimization" (with T. Georgiou), *Perspectives in Mathematical System Theory, Control, and Signal Processing*, Lecture Notes in Control and Information Sciences, volume 398, Springer-Verlag, 2010, pp. 321-331.

83. "Optimal mass transport for problems in control, statistical estimation, and image processing" (with E. Tannenbaum and T. Georgiou), to appear in a volume dedicated to 65th Birthday of J. William Helton, Birkhauser, 2012.

1 Introduction

The problem and need for robust visual tracking algorithms is widespread for both both military and civilian applications. Of particular relevance to the Air Force is tracking in tactical directed-energy engagements. Here one requires designating an aimpoint on complex, resolved targets in the presence of clutter, and maintaining the high-energy laser (HEL) on the aimpoint in the presence of target motion and atmospheric-induced aberrations. To date, tracking has been performed with 2D target images, but the recent development of angle-angle-range LADAR with high range resolution has enabled the possibility of using 3D target images for aimpoint selection and tracking. The visual tracking methodology developed in this work has proven to be very useful for such applications.

The overall technical approach uses a deterministic observer framework for visual tracking based on non-parametric implicit (level set) curve descriptions. The observer is continuous/discrete, with continuous-time system dynamics and discrete-time measurements. Its state space consists of an estimated curve position augmented by additional states (e.g., velocities) associated with every point on the estimated curve. Several simulation models may be used for state prediction, and measurements may be performed utilizing various segmentation techniques and optical flow computations. Special emphasis is given to the geometric formulation of the overall dynamical system. The discrete-time measurements lead to the problem of geometric curve interpolation and the discrete-time filtering of quantities propagated along with the estimated curve. Interpolation and filtering are intimately linked to the correspondence problem between curves. Correspondences are established by a Laplace equation approach. The proposed scheme is implemented completely implicitly (by Eulerian numerical solutions of transport equations) and thus naturally allows for topological changes and subpixel accuracy on the computational grid. It may be combined with geometric particle filtering as well as knowledge-based segmentation.

The geometric observer structure developed in this AFOSR sponsored research program is flexible enough to entertain the case where filtering position information is not utilized and may be replaced by static position measurements in case of clearly segmentable image data, leading to reduced order observers whose associated state information still needs to be filtered. Specifically, we have formulated the following novel techniques in the past three years:

(i) ***Geometric Particle Filtering for Visual Tracking***: Since filtering plays such an important role in our observer theory, and since we intend to use the geometric observers in conjunction with active contours, we have proposed a scheme that combines the advantages of particle filtering and geometric active contours realized via level set models for tracking deformable objects. We have investigated certain modifications to the standard particle filter (PF) [15, 3] as follows. First, we used an importance sampling (IS) density [3] which can be understood as an approximation to the optimal IS density when the optimal density is multi-modal. Next, we replaced IS by deterministic assignment when the variance of the IS density is very small (which occurs when the local deformation is small). Consequently, we are actually only sampling on the 6-dimensional space of affine deformations, while approximating local deformation by the mode of its posterior. This is what makes the our PF algorithm implementable in real time. (The full space of contour deformations is theoretically infinite.)

(ii) ***Information-Theoretic Approaches to Segmentation***: Segmentation is essential to our visual tracking framework. We therefore have developed a geometric contour based segmentation procedure that naturally fits into our geometric framework. We found that an active contour flow derived from an information-theoretic based criterion constituted a very reasonable approach in this regard. More specifically, we have proposed an active contour model whose evolution is driven by the gradient flow derived from an energy functional that is based on the Bhattacharyya distance. The approach can be viewed as a generalization of those segmentation methods, in which the active contours maximize the difference between a finite number of empirical moments of the distributions "inside" and "outside" the evolving contour. The model is very versatile and flexible since it allows one to easily accommodate a number of diverse image features. Further it can incorporate both local and global information.

2 Summary of Work

We summarize some of the key results developed as part of our AFOSR research program.

2.1 Geometric Observers

In [35], we formulated a general framework for geometric observer-like structures based on non-parametric implicit (level set) curve descriptions [38, 39]. Special emphasis was given to the geometric nature of the dynamical system as well as the key issue of robustness. In particular, we formulated an approach to the problem of information transport and filtering from a measurement curve to an estimated curve. We proposed a way to assess measurement reliability that allows for the selection of a local observer-gain. We should also note that the dynamical nature of the evolving curves described implicitly also allows for the observation of objects changing topology (i.e., objects breaking and merging during propagation) for which shape priors can be naturally incorporated.

This framework also fits in very naturally with geometric statistically based approaches for detection and identification; see, e.g. [61, 45, 46, 47] and the references therein. Indeed, one can apply the theory of Hidden Markov Models (HMM) for modelling the system dynamics and a particle filter to track the state as sketched in Section 2.2. Shape and motion parameters may be included in the hidden state vector (see our discussion below).

2.1.1 Observers for Tracking

The filtering of sensed data is a practical necessity when using the data to inform a feedback process. Real world signals derived from sensors have noise and disturbances that must be treated prior to incorporating the data into the feedback loop. Visual sensors are fundamentally different from traditional sensors (e.g., gyros, accelerometers, range sensors, GPS) in the sense that the true output for use in the feedback loop is usually not directly obtained from the sensor proper, but is extracted using some computer vision algorithm.

Filtering methodologies may be divided in three broad (not necessarily mutually exclusive) subcategories:

(1) **Pre-filtering**: Direct filtering of the image information obtained from the vision sensors, followed by the application of a computer vision algorithm.

(2) **Internal-state-filtering**: Filtering the internal states associated with a computer vision algorithm, based on the image information obtained from the vision sensors.

(3) **Post-filtering**: Direct application of a computer vision algorithm to the data obtained from the vision sensors, followed by the filtering of the output of the computer vision algorithm.

To illustrate the difference between these approaches, assume the objective is to find the centroid of a moving object given noisy image information from a vision sensor. For (1), the images are spatio-temporally filtered, the object is extracted from a filtered image, and the centroid is computed given the extracted object (the segmentation). For (3), the object is segmented from a static image, the centroid is computed, and the centroid position is filtered given the centroids from previous image frames. For (2), the object itself is modeled dynamically (system states being position, shape, velocity, etc.), the object states are filtered based on the spatio-temporal image information of the vision sensor, and the centroid is extracted from the internal state of the modeled object. All methods may use a model for the expected motion, resulting in model-based filtering methodologies. Post-filtering may be regarded as a subclass of internal-state-filtering, where the state-space is identical to the output space of the computer vision algorithm (e.g., the centroid position). In turn, this means that internal-state-filtering describes a richer class of systems.

Observers are internal-state-filters. They are a classical concept in control and estimation theory, where system states need to be reconstructed from measurement data. Examples include the classical deterministic Luenberger observer, the Kalman filter and its derivatives (the unscented Kalman filter, the extended

Kalman filter), as well as particle filters; see [48] and the references therein. They all share the common observer ingredients:

(O1) a dynamical state model and a measurement model of the system to be observed for state **prediction**,

(O2) a **measurement** methodology (e.g., a device to measure velocity, a thermometer, an object segmentation, etc.),

(O3) and an **error correction** scheme to reconcile measurement and prediction to form the state **estimation**.

Irrespective of the observer similarities (O1)-(O3), observer approaches differ in terms of the system and measurement class for which they are designed, and the estimation method being employed. System dynamics and system measurements may both be either discrete or continuous though the most common observers used in practice are either completely discrete or have continuous-time system dynamics and discrete-time measurements. In practice, measurements will be noisy and the system model and the measurement model will only be correct up to modelling errors ("modelling noise"). Observers need to be robust to such uncertainties. If noise processes (from modelling or measurement) are not neglected, system descriptions are based on stochastic differential or stochastic difference equations. In the most general setting an observer then becomes an estimation procedure for the conditional density function relating system states to the time history of measurements.

For problems in visual tracking, in order to make the observation problem tractable, we have proposed a novel observer design on the space of planar curves or surfaces in space (regarded as the boundaries of shapes), where the system model is continuous-time and the measurements are discrete-time. Special emphasis was given to a geometric formulation of the observer. The proposed approach may be viewed as a geometric filtering method for the class of computer vision algorithms using curve evolutions where the common observer building blocks (O1-O3) are reinterpreted in the context of dynamic curve evolution.

In our framework, by combining multiple measurements (e.g., shape based and non-shape based), we can assess the quality of measurements locally, and then locally adapt (e.g, a velocity error injection gain). We make extensive use of previous work on establishing correspondences between curves to transport measured quantities from the measurement to the evolving estimated curve.

2.1.2 Curves with Vector Fibers and State Evolution

We very briefly review some of the material on curve evolution theory here. We will be using closed planar curves to represent the boundaries of objects in this framework. The space of smooth closed planar curves, denoted by $C^\infty(S^1;\mathbb{R}^2)$, forms an infinite-dimensional manifold. When dealing with the evolution of curves, an additional temporal parameter is added to the curve description. In short, planar curve evolution may be described as the time-dependent mapping: $\mathcal{C}(p,t) : S^1 \times [0,\tau) \mapsto \mathbb{R}^2$, where $p \in [0,1]$ is the curve parameter, $\mathcal{C}(p,t) = [x(p,t),\ y(p,t)]^T$, and $\mathcal{C}(0,t) = \mathcal{C}(1,t)$. Define the interior and the exterior of a curve $\mathcal{C}$ on the domain $\Omega \subset \mathbb{R}^2$ as

$$int(\mathcal{C}) := \left\{ \boldsymbol{x} \in \Omega : (\boldsymbol{x} - \boldsymbol{x}_c)^T \mathcal{N} > 0,\ \forall\, \boldsymbol{x}_c \in \mathcal{C} \right\}, \text{ and } ext(\mathcal{C}) := \Omega \setminus \overline{int(\mathcal{C})},$$

where $\mathcal{N}$ is the unit inward normal to $\mathcal{C}$.

Properly embedding a manifold in a larger dimensional space avoids the need for parametric representations. Within the context of closed curves, $\mathcal{C}$ can be represented implicitly by a level set function $\Psi : \mathbb{R}^2 \times [0,\tau) \to \mathbb{R}$ [39], where

$$\Psi(0,t)^{-1} = trace(\mathcal{C}(\cdot,t)).$$

Frequently, Ψ is chosen to be a signed distance function. Given a curve evolution equation

$$\mathcal{C}_t = \boldsymbol{v},$$

where $\boldsymbol{v}$ is a velocity vector and subscripts denote partial derivatives, the corresponding level set evolution equation is

$$\Psi_t + \boldsymbol{v}^T \nabla \Psi = 0.$$

In order to define a dynamical model for curve evolution, the curve state space needs to be expanded to include curve velocities. The corresponding space is now a vector bundle. In brief, a vector bundle is a family of vector spaces parameterized by another space, in this case the space of closed planar curves. Locally, the curve vector bundle is diffeomorphic to the cross product of the space of closed curves, $C^\infty(S^1;\mathbb{R}^2)$, and a given model vector space, W. Typically, an element $\boldsymbol{w} \in W$ will be a vector-valued function defined on $trace(\mathcal{C}(\cdot,t))$. However, given an implicit representation for the curve and its vector bundle, a time-varying vector fiber element corresponding to the curve represented by Ψ is given by $\boldsymbol{w} : \mathbb{R}^2 \times [0,\tau) \to \mathbb{R}^2$, an appropriately extended version of the curve's fiber element.

When defining the evolution or deformation of a curve, the transport of the fiber quantities with the curve must also be defined. The transport of the fiber component, $\boldsymbol{w}$, in the implicit representation is induced by the curve evolution through the advection equation

$$\begin{cases} \boldsymbol{w}(\cdot,0) = \boldsymbol{w}_0, \\ \boldsymbol{w}_t + D\boldsymbol{w} \cdot \boldsymbol{v} = 0, \end{cases}$$

where $D\boldsymbol{w}$ denotes the Jacobian of $\boldsymbol{w}$, and $\boldsymbol{w}_0$ are the fiber quantities being propagated.

2.1.3 General Observer Structure

In the classical observer framework (e.g., as proposed by Luenberger [31]) there are prediction and measurement components. Prediction incorporates the dynamical assumptions made regarding the plant or, in the context of visual tracking, the movement of the object. In analogy with classical observer theory, our proposed observer structure will contain a prediction and a measurement part. To evolve the overall estimated curve, the prediction influence has to be combined with the measurement influence, leading to the correction step.

The observer to be defined is a continuous/discrete observer, i.e., the system evolves in continuous time with available measurements at discrete time instants $k \in \mathbb{N}_0^+$,

$$\begin{pmatrix} \mathcal{C} \\ \boldsymbol{w} \end{pmatrix}_t = \begin{pmatrix} \boldsymbol{v}(\mathcal{C},\boldsymbol{w},t) \\ \boldsymbol{f}(\mathcal{C},\boldsymbol{w},t) \end{pmatrix} + \boldsymbol{w}(t), \qquad \text{and} \qquad \boldsymbol{z}_k = \boldsymbol{h}_k\left(\begin{pmatrix} \mathcal{C} \\ \boldsymbol{w} \end{pmatrix}\right) + \boldsymbol{s}_k,$$

where $\boldsymbol{w}$ and $\boldsymbol{s}_k$ are the system and measurement noises, respectively, $\mathcal{C}$ represents the curve position, $\boldsymbol{w}$ denotes additional states transported along with $\mathcal{C}$ (e.g., velocities), and $(\cdot)_k$ denotes quantities given at discrete time points t_k.

The addition of a prediction model for the active contour, a measurement model for the active contour, and a correction step to the evolution and measurement described above form the general observer structure. The prediction and measurement models

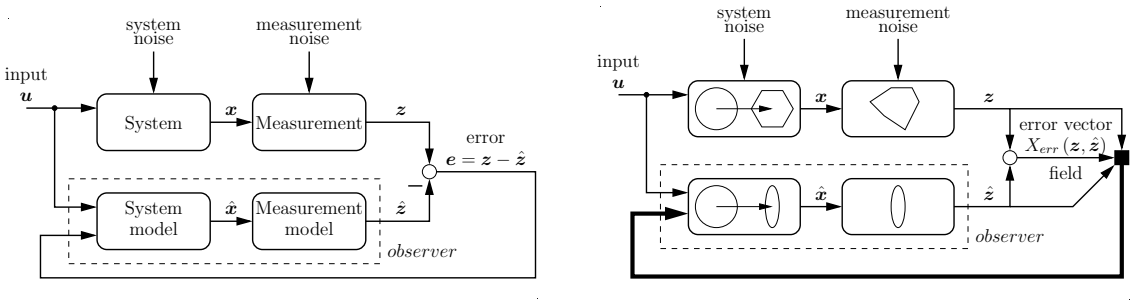
$$\begin{pmatrix} \hat{\mathcal{C}} \\ \hat{\boldsymbol{w}} \end{pmatrix}_t = \begin{pmatrix} \hat{\boldsymbol{v}}(\hat{\mathcal{C}},\hat{\boldsymbol{w}},t) \\ \hat{\boldsymbol{f}}(\hat{\mathcal{C}},\hat{\boldsymbol{w}},t) \end{pmatrix}, \qquad \hat{\boldsymbol{z}}_k = \hat{\boldsymbol{h}}_k\begin{pmatrix} \hat{\mathcal{C}} \\ \hat{\boldsymbol{w}} \end{pmatrix}, \tag{1}$$

simulate the true system dynamics and the measurement process, where the hat denotes simulated quantities.

For simplicity, consider the case when the complete state is measurable, $\boldsymbol{h}_k = \mathrm{id}$ (the identity map). The proposed continuous/discrete observer is

$$\begin{aligned} \begin{pmatrix} \hat{\mathcal{C}} \\ \hat{\boldsymbol{w}} \end{pmatrix}_t &= \begin{pmatrix} \hat{\boldsymbol{v}}(\hat{\mathcal{C}},\hat{\boldsymbol{w}},t) \\ \hat{\boldsymbol{f}}(\hat{\mathcal{C}},\hat{\boldsymbol{w}},t) \end{pmatrix}, \quad \hat{\boldsymbol{z}}_k = \begin{pmatrix} \hat{\mathcal{C}}_k \\ \hat{\boldsymbol{w}}_k \end{pmatrix}, \\ \begin{pmatrix} \hat{\mathcal{C}}_k(+) \\ \hat{\boldsymbol{w}}_k(+) \end{pmatrix} &= \Phi\left(X_{err}; \begin{pmatrix} \hat{\mathcal{C}}_k(-) \\ \hat{\boldsymbol{w}}_k(-) \end{pmatrix}, \begin{pmatrix} \mathcal{C}_k \\ \boldsymbol{w}_k \end{pmatrix}; \begin{pmatrix} K_k^{\mathcal{C}} \\ \boldsymbol{K}_k^{\boldsymbol{w}} \end{pmatrix}\right), \end{aligned} \tag{2}$$

where $(-)$ denotes the time just before a discrete measurement, $(+)$ the time just after the measurement, Φ is a correction function depending on the gain parameters $K_k^{\mathcal{C}}$ (a scalar) and $\boldsymbol{K}_k^{\boldsymbol{w}}$ (a matrix), and (X_{err}) is the error vector field. Figure 1 shows the observer structure as given in equation (2). In what follows, these observer components as applied to closed curves are described in further detail.



(a) Classical Euclidean Observer Structure.

(b) Symbolic Observer Structure for Curves.

Figure 1: *General Observer Structure.* In the Euclidean case, simple subtractions and additions are used for estimation error computations and state corrections. For curves, estimation errors are represented through an error vector field, relating the prediction to the measurement. This also requires the computation of a homotopy between the predicted and the measured curves. The error vector field X_{err} is computed through a correspondence procedure. To do error correction in the curve case requires the knowledge of the predicted measurement, the actual measurement, as well as the error vector field (denoted by the thick black input line to the system model). The measurement makes use of standard segmentation algorithms and the system model is given by a chosen prior. (For representational simplicity the observer structure figures show continuous time measurements. In the proposed discrete-time measurement case, the system state only gets updated at discrete time instants.)

2.1.4 Motion Priors

The prediction model part is a motion prior, describing the time-evolution of the closed curve, and possibly its vector fiber. It is problem dependent and should model as precisely as possible the dynamics of the object(s) to be tracked. Ideally the measurement part of an observer should only need to correct for inaccuracies due to noise. In practice it will be difficult (or even impossible) to provide an exact motion model so that the measurement part also needs to compensate for inaccuracies of the motion prior.

With this in mind, several priors are described in this section, increasing in complexity: the static prior, the constant velocity prior, the quasi-dynamic prior, and the dynamic elastic prior. These should be pointers to relatively general-purpose priors. They should be substituted by more accurate problem-specific priors, if available.

The simplest possible prior is the *static prior*, i.e., no motion at all, $\hat{\mathcal{C}}_t = 0$. For visual tracking the movement of a curve is then only driven by the measurement part of the observer. Were it not for the measurements, the curve would stay fixed at one position. The next simplest prior is the *constant velocity prior*, $\hat{\mathcal{C}}_{tt} = 0$.

Next, suppose that although the dynamics could not be predicted in closed form, the instantaneous velocity could be approximated or somehow estimated. In this case, the instantaneous velocity information could be used to propagate forward the curve,

$$\hat{\mathcal{C}}_t = (X_{est} \circ \mathcal{C} \cdot \mathcal{N})\mathcal{N},$$

where X_{est} is the estimated instantaneous velocity field. This is called the *quasi-dynamic prior*. One example of a quasi-dynamic prior would be to use the optical flow vector field as a motion prior. The optical flow field used could be computed using prior information or using current information. Technically, the latter suggestion is not a proper prior in our framework, since it implicitly depends on the current image through the optical flow calculation. Alternatively, any justifiable method resulting in a flow field for the closed curve could be used. The quasi-dynamic motion prior would be useful in cases where there is rich target motion and an available instantaneous model of flow for the visual information, but where a faithful model is not possible.

The *dynamic elastic prior* is based on the dynamic active curve described in [37] where the action integral $\mathcal{L} = \int_{t=t_0}^{t_1} \int_0^1 \left(\frac{1}{2}\mu \|\hat{\mathcal{C}}_t\|^2 - a \right) \|\hat{\mathcal{C}}_p\| \, dp \, dt$ is minimized. In contrast to the dynamic curve evolution of [37], a does not contain image information, but is a design parameter for curve regularization (it can either be a constant or a function over space and time), since the prior should not depend on underlying image information. The dynamic elastic prior for normal curve propagation is then given by:

$$\mu \hat{\mathcal{C}}_{tt} = \left(\frac{1}{2}\mu \|\hat{\mathcal{C}}_t\|^2 + a \right) \kappa \mathcal{N} - (\nabla a \cdot \mathcal{N}) \mathcal{N} - \frac{1}{2}\mu (\|\hat{\mathcal{C}}_t\|^2)_s \mathcal{T}. \tag{3}$$

If we give up our strict image independence of the prior and allow image influence as for the quasi-dynamic optical flow prior, a could become image dependent again and may be set to an image stopping function; this transforms equation (3) into the equation for the dynamic active curve [37].

Additional dynamic priors have been tested. These include dynamic priors that are area-preserving, length-preserving, smoothness-limiting, etc. Shape restrictions for dynamic priors could for example be accomplished by projecting the dynamic evolution onto a certain, specified shape equivalence class. Finally, the dynamical model may include additional states. These could for example be local estimates of state uncertainty, marker particles, etc, and would involve an expansion of the model vector space W.

2.1.5 Measurements

Measurements are used to drive the estimated model's states to the true system states. This is necessary, since generally the system model will be imperfect and the designed observer needs to be robust with respect to disturbances. For visual tracking, it is difficult to come up with accurate motion models, so simple approximations need to suffice.

The predicted measurements are based on the current state of the observer. Any of the standard segmentation algorithms can be used to come up with the "real" measurement that the predicted one has to be compared against to define the error measure.

This observer set-up has two crucial advantages:

- Any standard (static or dynamic) segmentation algorithm can be employed for the measurement. While the dynamic model is a model of a dynamically evolving curve, the measurement can utilize, for example, area-based or region-based segmentation algorithms.

- Static and dynamic approaches incorporating shape information exist [65]. If these approaches are used for the measurement curve, shape information can be introduced into the infinite dimensional model without the need for the explicit incorporation of the shape information into the dynamical model.

The inclusion of shape information as a constraint on the measurements contrasts with previous approaches aimed at including shape information into the dynamics of an evolving curve itself [7] where motion is restricted to affine motion. Using a finite dimensional motion group reduces the dimensionality of the evolution state space. Whereas a general curve evolution is infinite-dimensional, affine motion constraints lead to finite-dimensional descriptions and are thus relatively easy to implement and usually fast. To account for deviations from the motion model, correction terms need to be introduced, as is done in the deformotion work described in [65].

Position measurements can be accomplished by any static segmentation method, which may include shape information. Potential candidates are the classical geodesic active contour model [8, 26], or the Chan-Vese functional [9] (see our discussion in Section 2.2.5), or the information-based approach described in Section 2.3. Measurements of the vector fiber quantities are performed based on the location of the position measurements, if possible. In the case of dynamically evolving curves, velocities need to be measured on the measured curve via optical flow.

2.1.6 Error Correction

The observer framework proposed here requires a methodology to compare the predicted curve configuration to the measured curve configuration. Minimally, the comparison requires establishing unique correspondences between points on the measured and the predicted curve, e.g., defining a diffeomorphic homotopy between the two curves. The homotopy will be obtained through the flow of an error vector field defined between the two curves. In what follows, we describe the propagation of state information along the error vector field and the implicit computation of signed distance functions from which the curve correction homotopy is defined.

Error Vector Field

The error vector field to be defined is the manifold analogue to the measurement residual of an observer. Due to the geometry of the space of closed curves, there is no unique way to define the error vector field. In fact, one could consider the definition of the error vector field to be a design choice in setting up an observer for closed curves. Here, the method chosen is a Laplace equation based approach, whose error vector field induced flow is a diffeomorphism, which is easy to implement and fast to compute. Other approaches to define an error vector field exist, and in particular, diffeomorphic nonlinear registration methodologies [2] have been used as well and constitute a continuing research problem.

The variational formulation leading to the Laplace problem is

$$\min_u \int \|\nabla u\|^2 \, d\Omega, \quad \text{s.t.} \quad \text{trace}(\mathcal{C}_0) = u^{-1}(0), \ \text{trace}(\mathcal{C}_1) = u^{-1}(1), \tag{4}$$

where $\mathcal{C}_0$ is the source curve (the measurement curve) and $\mathcal{C}_1$ is the target curve (the predicted curve). Its solution requires careful construction of the interior and boundary conditions. The source curve and the target curve define the following solution domain decomposition of the total space Ω, $R := \overline{int(\mathcal{C}_0) \ominus int(\mathcal{C}_1)}$, $R_{pi} := int(\mathcal{C}_0) \cap int(\mathcal{C}_1)$, and $R_{lo} := \Omega \setminus \left(\overline{R \cup R_{pi}}\right)$, where $int(\mathcal{C})$ denotes the interior of the curve $\mathcal{C}$ and $\ominus$ is the set-symmetric difference.

To simplify the solution of (4) computationally, we change the boundary conditions to 0 for the interior curve parts $(\partial\overline{R_{pi}} \setminus (\mathcal{C}_0 \cap \mathcal{C}_1))$ and to 1 for the exterior curve parts $(\partial\left(R \cup R_{pi}\right))$. Note that both the exterior and the interior curve parts may be composed of subsets of $\mathcal{C}_0$ *and* $\mathcal{C}_1$ if $\mathcal{C}_0$ and $\mathcal{C}_1$ intersect. By changing the boundary conditions, we can compose a continuous solution function globally over all of the computational domain. The computed gradient field of this continuous solution may be reversed with respect to the original formulation (4), which can easily be accounted for after its computation. Thus the modified solution yields the same gradient field as the original formulation, which is the quantity of interest for the purpose of the observer design. Via the calculus of variations, a solution to (4) in the domain enclosed by the source and target curves *with modified boundary conditions* satisfies

$$\Delta u_s(\boldsymbol{x}) = 0, \quad \boldsymbol{x} \in R$$

with the boundary conditions

$$\begin{aligned} u_s(\boldsymbol{x}) &= 0, \ \boldsymbol{x} \in \partial\overline{R_{pi}} \setminus (\mathcal{C}_0 \cap \mathcal{C}_1), \\ u_s(\boldsymbol{x}) &= 1, \ \boldsymbol{x} \in \partial\left(R \cup R_{pi}\right), \end{aligned} \tag{5}$$

which is a simple reformulation of the minimization problem (4). To facilitate easy numerical computations of the error vector field, we extend the solution to the remainder of the image domain, by solving an additional Laplace equation on R_{lo} and a Poisson equation on R_{pi}:

$$\begin{aligned} \Delta u_{pi}(\boldsymbol{x}) &= c \quad \boldsymbol{x} \in \overline{R_{pi}}, \ c > 0 \\ \Delta u_{lo}(\boldsymbol{x}) &= 0, \quad \boldsymbol{x} \in \overline{R_{lo}}, \end{aligned}$$

with boundary conditions

$$\begin{aligned} u_{pi}(\boldsymbol{x}) &= 0,\ \boldsymbol{x} \in \partial\overline{R_{pi}}, \\ u_{lo}(\boldsymbol{x}) &= 1,\ \boldsymbol{x} \in \partial\left(R \cup R_{pi}\right), \\ u_{lo}(\boldsymbol{x}) &= 2,\ \boldsymbol{x} \in \partial\Omega. \end{aligned}$$

The combined solution

$$u(\boldsymbol{x}) = \begin{cases} u_{lo}(\boldsymbol{x}), & \boldsymbol{x} \in R_{lo}, \\ u_{pi}(\boldsymbol{x}), & \boldsymbol{x} \in R_{pi}, \\ u_s(\boldsymbol{x}), & \boldsymbol{x} \in R \end{cases}$$

defines the error vector field X_{err},

$$X_{err}(\boldsymbol{x}) := \begin{cases} \nabla u/\|\nabla u\|, & \boldsymbol{x} \in R, \\ \nabla u_o/\|\nabla u_o\|, & \boldsymbol{x} \in R_{lo}, \\ \nabla u_i/\|\nabla u_i\|, & \boldsymbol{x} \in R_{pi} \end{cases}$$

on Ω via the normalized gradient.

To illustrate the behavior of X_{err} assume the measured curve as $\mathcal{C}_0$ and the predicted curve as $\mathcal{C}_1$, where $\mathcal{C}_0$ is strictly interior of $\mathcal{C}_1$. Then X_{err} becomes a vector field flowing the measured curve into the predicted curve at unit speed. Flowing at unit speed implies that particles starting at C_0 and flowing according to X_{err} will reach C_1 at different times (proportional to the distance covered). The advantage of the Laplace based correspondence scheme is that it is fast, parametrization-free and allows for topological changes. The main disadvantage is that it may lead to unwanted correspondences since it is not invariant to translations, rotations or scale. Thus other correspondence methodologies will be considered in our upcoming research program [2].

Information Transport

The error vector field X_{err} defined above will be used to geometrically interpolate between two curves, thereby defining the curve correction homotopy $\Phi(X_{err}; \hat{\mathcal{C}}, \mathcal{C}; K^{\mathcal{C}})$ and also inducing a state correction homotopy $\Phi(X_{err}, (\hat{\mathcal{C}}, \hat{q}), (\mathcal{C}, q); K)$. Geometric interpolation is achieved by measuring the distance between correspondence points along the characteristics, defined by the error vector field, that connect them and subsequent flow up to a certain percentage of this distance. Performing this procedure entails solving a series of associated transport equations.

Given that the curve evolution is performed implicitly, preserving this feature of the algorithm requires implicit information transport between the measured and the predicted curves. Consider the advection equation with source term s and velocity field X,

$$x_t + \boldsymbol{X}^T \nabla x = s. \tag{6}$$

The characteristic curves, $c'(t) = X \circ c(t)$, satisfy the ordinary differential equation

$$\frac{d}{dt} x(c(t), t) = s.$$

Thus, to advect vector quantities w along a velocity field X, solve

$$\begin{cases} w(\cdot, 0) = w_0, \\ w_t + Dw \cdot X = 0, \end{cases} \tag{7}$$

where w_0 denotes the given vector quantities w at time 0, i.e., the initial conditions of the partial differential equation (7), and Dw denotes the Jacobian of w. The vector quantity w may include for example local

velocities. To be able to interpolate curve positions, which will subsequently be used to define the curve correction homotopy given below, distances between curves need to be defined. Since Euclidean distances may not be desirable for complicated curve shapes, the proposed observer framework uses a distance measure induced by a flow field X (here, the error vector field X_{err}). Given a flow field X and a particle p initially located at $\boldsymbol{x}_0$, its travelling distance at position $\boldsymbol{x}$ is defined as the arc-length of the characteristic curve, whose tangents are aligned with X, connecting $\boldsymbol{x}_0$ and $\boldsymbol{x}$. To measure traveling distances from a complete set of initial locations, as specified by $d^{-1}(\cdot, 0)$, solve

$$\begin{cases} d(\cdot, 0) = 0, \\ d_t + \frac{1}{\|X\|} X^T \nabla d = 1, \end{cases} \tag{8}$$

where $X \neq 0$ is assumed. Equations (6)-(8) are Hamilton-Jacobi equations. Efficient numerical methods exist to compute steady-state solutions of Hamilton-Jacobi equations [24].

Curve Correction Homotopy

When $\hat{\Psi}$ and Ψ implicitly represent the curves $\hat{\mathcal{C}}$ and $\mathcal{C}$ the interpolation can be accomplished implicitly to subpixel accuracy. In order to determine the travelling distance from each curve to the other along X_{err}, compute

$$\begin{aligned} \hat{d}_t + S(\hat{\Psi}) X_{err}^T \nabla \hat{d} &= S(\hat{\Psi}), \qquad \hat{d}(\boldsymbol{x}, 0) = \hat{\Psi}, \\ (d_m)_t + S(\Psi) X_{err}^T \nabla d_m &= S(\Psi), \qquad d_m(\boldsymbol{x}, 0) = \Psi, \end{aligned} \tag{9}$$

where

$$S(x) := \begin{cases} 0, & \text{if } \|x\| \leq 1, \\ \frac{x}{\sqrt{\epsilon + x^2}}, & \text{otherwise.} \end{cases}$$

Here, $S(\boldsymbol{x})$ denotes a smoothed sign function, which allows for the bidirectional measurement of traveling distance, along X_{err} (resulting in positive distance values) and in the opposite direction of X_{err} (resulting in negative distance values). The results may be interpreted as signed distance level set functions warped with respect to the error vector field X_{err}. The distance error functions, $\hat{d}$ and d_m, obtained by solving the equation (9) are interpolated to yield the interpolated distance function d_i,

$$d_i = (1 - \alpha)\hat{d} + \alpha d_m, \; w \in [0, 1],$$

which subsequently gets redistanced according to

$$(\hat{\Psi}_i)_t + S(\hat{\Psi}_i^0) \|\nabla \hat{\Psi}_i\| = S(\hat{\Psi}_i^0), \qquad \hat{\Psi}_i(\boldsymbol{x}, 0) = d_i,$$

arriving at the corrected distance function $\hat{\Psi}_i$, The weighting factor α geometrically interpolates the estimated and the measured curve.

Vector Fiber Transport

In order to compare and correct the vector fiber quantities, they need to be transported to the new interpolated curve, located somewhere within the region R. For the measured quantities $\boldsymbol{w}_m$ and for the estimated quantities $\hat{\boldsymbol{w}}$ utilize the corresponding transport equations

$$\begin{aligned} (\boldsymbol{p}_m)_t + S(\Psi) X_{err}^T \nabla(\boldsymbol{p}_m) = 0, \quad \boldsymbol{p}_i(\boldsymbol{x}, 0) = \boldsymbol{w}_i, \\ (\hat{\boldsymbol{p}})_t + S(\hat{\Psi}) X_{err}^T \nabla \hat{\boldsymbol{p}} = 0, \quad \hat{\boldsymbol{p}}(\boldsymbol{x}, 0) = \hat{\boldsymbol{w}}, \end{aligned}$$

to propagate the values throughout the domain.

Performing the Error Correction

The error correction scheme builds on the framework described above. We assume that the correction function can be written as

$$\Phi\left(X_{err};\begin{pmatrix}\hat{\mathcal{C}}_k(-)\\ \hat{\boldsymbol{w}}_k(-)\end{pmatrix},\begin{pmatrix}\mathcal{C}_k\\ \boldsymbol{w}_k\end{pmatrix};\begin{pmatrix}K_k^{\mathcal{C}}\\ \boldsymbol{K}_k^{\boldsymbol{w}}\end{pmatrix}\right)=\begin{pmatrix}\Phi^{\mathcal{C}}\left(X_{err};\hat{\mathcal{C}}_k(-),\mathcal{C}_k;K_k^{\mathcal{C}}\right)\\ \Phi^{\boldsymbol{w}}\left(X_{err};\begin{pmatrix}\hat{\mathcal{C}}_k(-)\\ \hat{\boldsymbol{w}}_k(-)\end{pmatrix},\begin{pmatrix}\mathcal{C}_k\\ \boldsymbol{w}_k\end{pmatrix};\boldsymbol{K}_k^{\boldsymbol{w}}\right)\end{pmatrix}.$$

The error correction for the curve position is then

$$\hat{\mathcal{C}}_k(+)=\Phi^{\mathcal{C}}\left(X_{err};\hat{\mathcal{C}}_k(-),\mathcal{C}_k;K_k^{\mathcal{C}}\right),$$

which amounts to curve interpolation and gets computed as

$$\text{trace}(\hat{\mathcal{C}}_k(+))=\hat{\Psi}_i(0)^{-1},\quad K_k^{\mathcal{C}}=\alpha=\alpha_k.$$

State information needs to be exchanged and compared between the measured and the estimated curves. Further, the final filtering results needs to be associated with $\hat{\mathcal{C}}_k(+)$. This is accomplished by the error correction for the vector fiber

$$\hat{\boldsymbol{w}}_k(+)=\Phi^{\boldsymbol{w}}\left(X_{err};\begin{pmatrix}\hat{\mathcal{C}}_k(-)\\ \hat{\boldsymbol{w}}_k(-)\end{pmatrix},\begin{pmatrix}\mathcal{C}_k\\ \boldsymbol{w}_k\end{pmatrix};\boldsymbol{K}_k^{\boldsymbol{w}}\right)$$

which amounts to point-wise filtering computed as

$$(\hat{\boldsymbol{w}}_i)_k(+)=(\hat{\boldsymbol{p}}_i)_k+(\boldsymbol{K}_{ij})_k^{\boldsymbol{w},\boldsymbol{w}}\left((\boldsymbol{p}_j)_k-(\hat{\boldsymbol{p}}_j)_k\right)+(\boldsymbol{K}_i)_k^{\boldsymbol{w},\mathcal{C}}\left(d_m-\hat{d}\right)$$

and evaluated at $\hat{\mathcal{C}}_k(+)$, where repeated indices are summed over and $\boldsymbol{K}_k^{\boldsymbol{w}}$ is assumed to be block-diagonal and decomposes into a gain matrix for the fiber quantities ($\boldsymbol{K}_k^{\boldsymbol{w},\boldsymbol{w}}$) and for the curve position error ($\boldsymbol{K}_k^{\boldsymbol{w},\mathcal{C}}$). Table 1 gives a description of the overall geometric observer algorithm.

Finally, we should note that the observer described here is a full-order observer, that is, all states are observed.

2.1.7 Other Results on Geometric Observers

We have developed a geometric design methodology for observing or filtering dynamically evolving curves over time. State measurements are performed via static segmentations, making the framework flexible and powerful, since any kind of curve segmentation may be used for the measurement step, including curve segmentations incorporating shape information. In this way, measurements may be used to induce shape information to the estimated curve without the need for explicit incorporation of shape information into the motion prior. In comparison to most previous approaches there is no finite dimensional motion model. The observer framework is geometric,leading to geometrically meaningful observer gains. Further, the overall methodology extends to closed hypersurfaces of codimension one which represent the boundaries of 3D shapes and has been carried out in the past year as part of our AFOSR work. In our AFOSR research, we have also introduced statistical information that requires the notion of a mean shape of curves and curve covariances as well as adaptive observer gains based on curve statistics.

2.2 Geometric Particle Filtering

In conjunction with the geometric observer theory, we have developed a geometric particle filter framework in [45, 46, 47].

Geometric observer algorithm:

repeat

1) *Propagate curve* under the prediction model (Section 2.1.4) for the time-span
between two image measurements (usually given by the camera frame rate).
Initial conditions are given by the current observer state.

2) *Obtain curve measurements* by image segmentation, optical flow, etc. (Section 2.1.5).

3) *Reconcile internal observer state with the measurements by error correction* (Section 2.1.6):
a) Establish the error vector field (to induce correspondences; Section 2.1.6).
b) Flow measurements and observer states along the error vector field (Section 2.1.6).
c) Perform update of the internal observer state (Section 2.1.6).

until end of tracking sequence

Table 1: Description of the geometric observer algorithm.

The overall algorithm is based on three modifications of the standard particle filter (PF) [15, 3]: (i) First, we utilize an importance sampling (IS) density [3] which can be considered as an approximation to the optimal IS density when the optimal density is multi-modal. (ii) We replace IS by deterministic assignment when the variance of the IS density is very small. *Because of this step, we are actually only sampling on the 6-dimensional space of affine deformations, while approximating the local deformation by the mode of its posterior. This is what makes our PF algorithm implementable in real time.* (iii) Further, we proposed a technique for the computation of an approximation to the mode of the posterior of the local deformation. As explained in [45, **?**, 47], these modifications are useful to reduce computational complexity of any large dimensional state tracking problem.

2.2.1 The Particle Filtering Algorithm

This section describes the basics of the proposed method. Let $\mathcal{C}_t$ denote the contour at time t ($\mathcal{C}_t$ is represented as the zero level set of $\Psi_t(x)$, i.e. $\mathcal{C}_t = \{x \in \mathbb{R}^2 : \Psi_t(x) = 0\}$ [39]), and A_t denote a 6-dimensional affine parameter vector with the first 4 parameters representing rotation, skew and scale, respectively, and the last 2 parameters representing translation. *Note that in this section (Section 2.2) and in Section 2.3 below, we let $\Psi_t(x) := \Psi(x(t), t)$. Previously, the subscript t denoted partial derivative.*

We employ the affine parameters (A_t) and the contour ($\mathcal{C}_t$) as the state, i.e., $X_t = [A_t, \mathcal{C}_t]$ and treat the image at time t as the observation, i.e., $Y_t = \text{Image}(t)$. Denote by $Y_{1:t}$ all the observations until time t. Particle filtering [15] allows for recursively estimating $p(X_t|Y_{1:t})$, the posterior distribution of the state given the prior $p(X_{t-1}|Y_{1:t-1})$. We utilize the basic theory of particle filtering here as described in [15]. The general idea behind the proposed algorithm is as follows:

- **Importance Sampling:** Predict the affine parameters A_t (parameters governing the rigid motion of the object) and perform importance sampling for $\mathcal{C}_t$ to obtain local deformation in shape, i.e.,
  - Generate samples $\{A_t^{(i)}, \mu_t^{(i)}\}_{i=1}^N$ using:
  - Perform L steps of curve evolution on each $\mu_t^{(i)}$:

$$\mathcal{C}_t^{(i)} = f_{CE}(\mu_t^{(i)}, Y_t, u_{t,def}^{(i)}), \quad u_{t,def}^{(i)} \sim \mathcal{N}(0, \Sigma_{def}) \ .$$

- **Weighting and Resampling:** Calculate the importance weights and normalize [15], i.e.,

$$\tilde{w}_t^{(i)} = \frac{p(Y_t|X_t^{(i)})\; p(X_t^{(i)}|X_{t-1}^{(i)})}{q(X_t^{(i)}|X_{t-1}^{(i)}, Y_t)} \propto \frac{e^{\frac{-E_{image}(Y_t, \mathcal{C}_t^{(i)})}{\sigma_{obs}^2}}\; e^{\frac{-d^2(\mathcal{C}_t^{(i)}, \mu_t^{(i)})}{\sigma_d^2}}}{\mathcal{N}(f_{CE}(\mu_t^{(i)}, Y_t), \Sigma_{def})}, \quad w_t^{(i)} = \frac{\tilde{w}_t^{(i)}}{\sum_{j=1}^{N} \tilde{w}_t^{(j)}}\;,$$

  where d^2 is any distance metric between shapes (see Section 2.2.7) and E_{image} is any image based energy functional (see Section 2.2.5). Resample to generate N particles $\{A_t^{(i)}, \mathcal{C}_t^{(i)}\}$ distributed according to $p(A_t, \mathcal{C}_t|Y_{1:t})$. The resampling step improves sampling efficiency by eliminating particles with very low weights. We now briefly explain in detail each of the preceding steps.

2.2.2 The System and Observation Model

The problem of tracking deforming objects can be separated into two parts: a) Tracking the global rigid motion of the object; b) Tracking local deformations in the shape of the object, which can be defined as any departure from rigidity (non-affine deformations). The global motion (affine transformation) can be modeled by the 6 parameters of an affine transformation, A_t, using a first order Markov process. We assume that the local deformation from one frame to the next is small and can be modeled by deformation in the shape of the contour $\mathcal{C}_t$. Thus, the state vector is given by $X_t = [A_t, \mathcal{C}_t]$. The system dynamics based on the above assumption can be written as:

$$\begin{aligned} A_t &= f_p A_{t-1} + u_t, \;\; u_t \sim \mathcal{N}(0, \Sigma_A), \\ \hat{x} &= \begin{bmatrix} A_{t,1} & A_{t,2} \\ A_{t,3} & A_{t,4} \end{bmatrix} x + \begin{bmatrix} A_{t,5} \\ A_{t,6} \end{bmatrix}, \; \forall x \in \mathcal{C}_{t-1}, \;\; \hat{x} \in \mu_t, \;\; \mu_t := A_t(\mathcal{C}_{t-1}), \\ \mathcal{C}_t &= f_{def}(\mu_t, u_{t,def}), \quad u_{t,def} \sim \mathcal{N}(0, \Sigma_{def}), \end{aligned} \tag{10}$$

where f_p models global rigid motion of the object while f_{def} is a function that models the local shape deformation of the contour.

We further assume that the likelihood probability, i.e., probability of the observation $Y_t = \text{Image}(t)$ given state X_t, is defined by $p(Y_t|X_t) = p(Y_t|\mathcal{C}_t) \propto e^{\frac{-E_{image}(\mathcal{C}_t, Y_t)}{\sigma_{obs}^2}}$, where E_{image} is any image dependent energy functional and σ_{obs}^2 is a parameter that determines the shape of the pdf (probability density function). The normalization constant in the above definition has been ignored since it only affects the scale and not the shape of the resulting pdf.

In general, it is not easy to predict the shape of the contour at time t (unless the shape deformations are learned *a priori*) given the previous state of the contour at time $t-1$, i.e., it is not easy to find a good function f_{def} that can model the shape deformations and allows one to sample from an infinite (theoretically) dimensional space of curves. Thus, it is very difficult to draw samples for $\mathcal{C}_t$ from the prior distribution. This problem can be solved by doing importance sampling [10], and is one of the main motivations for doing curve evolution as explained in the following sections. Thus, samples for A_t can be obtained by sampling from $\mathcal{N}(f_p A_{t-1}, \Sigma_A)$ while samples for $\mathcal{C}_t$ are obtained using importance sampling, i.e., we perform importance sampling only on part of the state space. This technique of using importance sampling allows for obtaining samples for $\mathcal{C}_t$ using the latest observation (image) at time t [59].

The central idea behind importance sampling [10] is as follows: Suppose $p(x) \propto q(x)$ is a probability density from which it is difficult to draw samples and $q(x)$ is a density (proposal density or importance density) which is easy to sample from, then an approximation to $p(\cdot)$ is given by $p(x) \approx \sum_{i=1}^{N} w^i \delta(x - x^i)$, where $w^i \propto \frac{p(x^i)}{q(x^i)}$ is the normalized weight of the i-th particle. So, if the samples, $X_t^{(i)}$, were drawn from an importance density, $q(X_t|X_{1:t-1}, Y_{1:t})$, and weighted by $w_t^{(i)} \propto \frac{p(X_t^{(i)}|Y_{1:t})}{q(X_t^{(i)}|X_{1:t-1}^{(i)}, Y_{1:t})}$, then $\sum_{i=1}^{N} w_t^{(i)} \delta(X_t^{(i)} - X_t)$ approximates $p(X_t|Y_{1:t})$.

In this proposal, the state is assumed to be a hidden Markov process, i.e.,

$$p(X_t|X_{1:t-1}) = p(X_t|X_{t-1}), p(Y_t|X_{1:t}) = p(Y_t|X_t),$$

and we further assume that the observations are conditionally independent given the current state, i.e. $p(Y_{1:t}|X_{1:t}) = \prod_{\tau=1}^{t} p(Y_\tau|X_\tau)$. Furthermore, if the importance sampling density is assumed to depend only on the previous state X_{t-1} and current observation Y_t, we get $q(X_t|X_{1:t-1}, Y_{1:t}) = q(X_t|X_{t-1}, Y_t)$. This gives the following recursion for the weights [10]: $w_t^{(i)} = w_{t-1}^{(i)} \frac{p(Y_t|X_t^{(i)})p(X_t^{(i)}|X_{t-1}^{(i)})}{q(X_t^{(i)}|X_{t-1}^{(i)},Y_t)}$. The importance density $q(.)$ and the prior density $p(.)$ can now be written as

$$q(X_t|X_{t-1}, Y_t) = p(A_t|A_{t-1})\; q(\mathcal{C}_t|\mu_t, Y_t), \quad p(X_t|X_{t-1}) = p(A_t|A_{t-1})\; p(\mathcal{C}_t|\mu_t), \tag{11}$$

where $q(A_t|A_{t-1}) = p(A_t|A_{t-1})$, since A_t is sampled from $p(A_t|A_{t-1}) = \mathcal{N}(f_p A_{t-1}, \Sigma_A)$. Thus, the weights can be calculated from:

$$w_t^{(i)} = w_{t-1}^{(i)} \; \frac{p(Y_t|X_t^{(i)})\; p(\mathcal{C}_t^{(i)}|\mu_t^{(i)})}{q(\mathcal{C}_t^{(i)}|\mu_t^{(i)}, Y_t)} \; . \tag{12}$$

The probability $p(\mathcal{C}_t|\mu_t)$ can be calculated using any suitable measure of similarity between shapes (modulo a rigid transformation). One such measure is to take $p(\mathcal{C}_t|\mu_t) \propto e^{\frac{-d^2(\mathcal{C}_t,\mu_t)}{\sigma_d^2}}$, where σ_d is assumed to be very small such that it satisfies the constraint of (10) in [44] and d^2 is any metric on the space of closed curves. In this approach, we employ the distance measure given in Section 2.2.7.

2.2.3 Approximating the Optimal Importance Density

The choice of the importance density is a critical design issue for implementing a successful particle filter. The proposal distribution $q(\cdot)$ should be such that particles generated by it, lie in the regions of high observation likelihood. One way of doing this is to use a proposal density which depends on the current observation [59]. The optimal importance density (one that minimizes the variance of the weights conditioned on X_{t-1} and Y_t) has been shown to be $p(X_t|X_{t-1}, Y_t)$. But in many cases, it cannot be computed in closed form. For unimodal posteriors, it can be approximated by a Gaussian with mean given by its mode, which is also equal to the mode of $p(Y_t|X_t)\; p(X_t|X_{t-1})$. In our case, the distribution $p(A_t|A_{t-1})$ can be multi-modal, and hence we have employed the following: Sample A_t from the prior state transition kernel, $p(A_t|A_{t-1})$, and find the mode of $p(Y_t|X_t)\; p(\mathcal{C}_t|\mu_t)$ to obtain samples for $\mathcal{C}_t$. Notice that, for small deformations, $p(Y_t|X_t)\; p(\mathcal{C}_t|\mu_t)$ is indeed unimodal [44]. Using (11) and the likelihood probability $p(Y_t|X_t)$ defined before, finding the mode of $p(Y_t|X_t)\; p(\mathcal{C}_t|\mu_t)$ is equivalent to finding the minimizer of

$$E_{tot}(\mathcal{C}_t, \mu_t, Y_t) = \frac{E_{image}(\mathcal{C}_t, Y_t)}{\sigma_{obs}^2} + \frac{d^2(\mathcal{C}_t, \mu_t)}{\sigma_d^2}.$$

Notice that from this energy point of view, it is clear why we can ignore the partition constants (in the definition of $p(Y_t|\mathcal{C}_t)$ and $p(\mathcal{C}_t|\mu_t)$) which are needed to normalize the various densities so that they define proper probability measures. Indeed, we are only interested in the minimizer of E_{tot}.

Finding the exact minimizer of E_{tot} for each particle at each t is computationally expensive and hence we use the following approximation: Assuming a small deformation between $t-1$ and t, both the terms in this summation will be locally convex (in the neighborhood of the minimizers of both terms), and so the minimizer of the sum will lie between the individual minimizers of each term. Thus, an approximate solution to find the minimum of E_{tot} will be to start from the minimizer of one term and go a certain distance (i.e., a certain number of iterations of gradient descent) towards the minimizer of the second. It is easy to see that $\mathcal{C} = \mu_t$ minimizes the second term, and hence, starting with μ_t as the initial guess for $\mathcal{C}$, and performing L iterations of gradient descent will move $\mathcal{C}$ a given distance towards the minimizer of

E_{image}, where L is chosen experimentally. We would like to reiterate here that the optimal choice of L will be one that finds a curve C to minimize E_{tot}, but to avoid performing the complete minimization of E_{tot}, we are doing this approximation, and have found that it works well in practice.

Using the above technique, we are actually only sampling on the 6-dimensional space of affine deformations, while approximating local deformation by the mode of its posterior. In general, the "mode tracker" method described above reduces the computations significantly.

2.2.4 Curve Evolution for Computing $\mathcal{C}_t$

We now describe how to obtain samples for $\mathcal{C}_t$ by doing gradient descent on the energy functional E_{image}. This operation is represented by the function f_{CE}. The non-linear function $f_{CE}(\mu, Y, u_{def})$ is evaluated as follows (for $k = 1, 2, ..., L$):

$$\mu^0 = \mu, \quad \mu^k = \mu^{k-1} - \alpha^k \nabla_\mu E_{image}(\mu^{k-1}, Y, u_{def}), \quad f_{CE}(\mu, Y, u_{def}) = \mu^L \ . \tag{13}$$

The above equation is basically a PDE which moves an initial guess of the contour so that E_{image} is minimized. Note that $u_{def} \sim \mathcal{N}(0, \Sigma_{def})$ is a noise vector that is added to the "velocity" of the deforming contour at each point $x \in \mu$ (see [39, 25] for details on how to evolve a contour using level set representation). For practical examples with small deformations, Σ_{def} is very small and in fact, even when one does not add any noise to f_{CE}, there is no noticeable change in performance. In numerical experiments, we have not added any noise to the curve evolution process. Thus, the importance sampling density for A_t is $p(A_t|A_{t-1})$ while that for $\mathcal{C}_t$ is $q(\mathcal{C}_t|\mu_t, Y_t) = \mathcal{N}(f_{CE}(\mu_t, Y_t), \Sigma_{def} \to 0)$. The curve $\mathcal{C}_t$ thus obtained incorporates the prediction for global motion and local shape deformation.

Alternative Interpretation for L-Iteration Gradient Descent

We perform only L iterations of gradient descent since we do not want to evolve the curve until it reaches a minimum of the energy, E_{image}. Evolving to the local minimizer is not desirable since the minimizer would be independent of all starting contours in its domain of attraction and would only depend on the observation, Y_t. Thus the state at time t would loose its dependence on the state at time $t-1$, and this may cause loss of track in cases where the observation is bad. In effect, choosing L to be too large (taking the curve very close to the minimizer) can move all the samples too close to the current observation and thus result in reduction of the variance of the samples leading to "sample degeneracy." At the same time, if L is chosen to be too small, the particles will not be moved to the region of high observation likelihood and this can lead to "sample impoverishment." The choice of L depends on how much one trusts the system model versus the obtained measurements. Note that, L will of course also depend on the step-size of the gradient descent algorithm as well as the type of PDE used in the curve evolution equation.

Based on the above discussion, the importance weights in (12) can be calculated as follows:

$$\begin{aligned} w_t^{(i)} = w_{t-1}^{(i)} \frac{p(Y_t|X_t^{(i)})\, p(\mathcal{C}_t^{(i)}|\mu_t^{(i)})}{q(\mathcal{C}_t^{(i)}|\mu_t^{(i)}, Y_t)} &\propto w_{t-1}^{(i)} \frac{e^{\frac{-E_{image}(\mathcal{C}_t^{(i)}, Y_t)}{\sigma_{obs}^2}} e^{\frac{-d^2(\mathcal{C}_t^{(i)}, \mu_t^{(i)})}{\sigma_d^2}}}{\mathcal{N}(f_{CE}(\mu_t^{(i)}, Y_t), \Sigma_{def})} \\ &\propto w_{t-1}^{(i)} \exp\left(\frac{-E_{image}(\mathcal{C}_t^{(i)}, Y_t)}{\sigma_{obs}^2}\right) \exp\left(\frac{-d^2(\mathcal{C}_t^{(i)}, \mu_t^{(i)})}{\sigma_d^2}\right), \end{aligned} \tag{14}$$

where we have used the fact that $\mathcal{C}_t^{(i)}$ is the mean and Σ_{def} is very close to zero, implying that $\mathcal{N}(\mathcal{C}_t^{(i)}, \Sigma_{def} \to 0)$ can be approximated by a constant for all particles.

2.2.5 Curve Evolution using Chan-Vese model

Many methods have been proposed which incorporate geometric and/or photometric (color, texture, intensity) information in order to segment images robustly in presence of noise and clutter. In our case, in

the prediction step above, f_{CE} can be any edge-based or region-based curve evolution equation. In our preliminary work, the Mumford-Shah functional [34] as modelled by Chan and Vese was used [9] to obtain the curve evolution equation. In Section 2.3, we will describe a far-reaching generalization of this segmentation methodology that will allow us to track in much more complex noisy environments.

For the Chan-Vese model [9], one applies the calculus of variations to minimize the following energy E_{image}:

$$E_{image} = \int_{\Omega} (I - c_1)^2 H(\Psi) dx\, dy + \int_{\Omega} (I - c_2)^2 (1 - H(\Psi))\, dx\, dy + \nu \int_{\Omega} |\nabla H(\Psi)| dx\, dy\,, \quad (15)$$

where c_1,c_2 and the Heaviside function $H(\Psi)$ are defined as

$$c_1 = \frac{\int I(x,y) H(\Psi) dx\, dy}{\int H(\Psi) dx\, dy}, \quad c_2 = \frac{\int I(x,y)(1 - H(\Psi)) dx\, dy}{\int (1 - H(\Psi)) dx\, dy}, \quad H(\Psi) = \begin{cases} 1 & \Psi \geq 0, \\ 0 & \text{else}, \end{cases}$$

and finally $I(x,y)$ is the image and Ψ is the level set function. The energy E_{image} can be minimized by doing gradient descent via the following PDE [9, 34]:

$$\frac{\partial \Psi}{\partial \tau} = \delta_\epsilon(\Psi) \left[\nu \,\text{div}\left(\frac{\nabla \Psi}{|\nabla \Psi|} \right) - (I - c_1)^2 + (I - c_2)^2 \right], \text{where} \quad \delta_\epsilon(s) = \frac{\epsilon}{\pi(\epsilon^2 + s^2)}\,,$$

where τ is the evolution time parameter and the contour $\mathcal{C}$ is the zero level set of Ψ.

2.2.6 Dealing with Multiple Objects

In principle, the CONDENSATION filter [7] could be used for tracking multiple objects. The posterior distribution will be multi-modal with each mode corresponding to one object. However, in practice it is very likely that a peak corresponding to the dominant likelihood value will increasingly dominate over all other peaks when the estimation progresses over time. In other words, a dominant peak is established if some objects obtain larger likelihood values more frequently. So, if the posterior is propagated with fixed number of samples, eventually, all samples will be around the dominant peak. This problem becomes more pronounced in cases where the objects being tracked do not have similar photometric or geometric properties. One may deal with this issue using the method in [57] by first finding the clusters within the state density to construct a Voronoi tessalation [51] and then resampling within each Voronoi cell separately. Other solutions proposed by [54] have also been tested for multiple object tracking.

2.2.7 Occlusions

Prior shape knowledge is necessary when dealing with occlusions. In particular, in [66], the authors incorporate "shape energy" in the curve evolution equation to deal with occlusions. Any such energy term can be used in the proposed model to deal with occlusions. In numerical experiments, we have dealt with this issue in a slightly different way by incorporating the shape information in the weighting step instead of the curve evolution step, i.e., we calculate the likelihood probability for each particle using the image energy E_{image} (15) and a shape dissimilarity measure d^2 as follows:

$$p(Y_t | X_t^{(i)}) = \lambda_1 \left(\frac{e^{\frac{-E_{image}^{(i)}}{\sigma_{obs}^2}}}{\sum_{j=1}^{N} e^{\frac{-E_{image}^{(j)}}{\sigma_{obs}^2}}} \right) + \lambda_2 \left(1 - \frac{d^2(\Psi^{(s)}, \Psi^{(i)})}{\sum_{j=1}^{N} d^2(\Psi^{(s)}, \Psi^{(j)})} \right)\,, \quad (16)$$

where $\lambda_1 + \lambda_2 = 1$, and $d^2(\Psi^{(s)}, \Psi^{(i)})$ is the dissimilarity measure (modulo a rigid transformation) given by, $d^2(\Psi^{(s)}, \Psi^{(i)}) = \int_\Omega (\Psi^{(s)} - \Psi^{(i)})^2 \, \frac{h(\Psi^{(s)}) + h(\Psi^{(i)})}{2}\, dx\, dy$, with $h(\Psi) = \frac{H(\Psi)}{\int_\Omega H(\Psi)\, dx\, dy}$, where $\Psi^{(s)}$

and $\Psi^{(i)}$ are the level set functions of a template shape and the i-th contour shape, respectively. The dissimilarity measure gives an estimate of how different two given shapes (in particular, their corresponding level sets) may be. So, higher values of d^2 indicate more dissimilarity in shape. This strategy may be justified by noting that in case of occlusion, E_{image} will be higher for a contour that encloses the desired region compared to a contour that excludes the occlusion. Since particle weights are a function of E_{image}, the MAP estimate will be a particle that is not the desired shape. However, using the weighting scheme proposed above, particles which are closer to the template shape are more likely to be chosen than particles with "occluded shapes" (i.e., shapes which include the occlusion).

2.3 Information-Theoretic Approach to Segmentation

The choice of the segmentation algorithm is very important to the success of our approach to visual tracking. We have considered the Chan-Vese model which only uses the first moment for segmentation for the geometric particle filtering just considered. In our AFOSR work, we have developed a far-reaching generalization employing all of the statistical moments [33, 29].

More precisely, we address the problem of image segmentation by means of active contours, whose evolution is driven by the gradient flow derived from an energy functional that is based on the Bhattacharyya distance. Because of the broad statistical nature of the flow, we believe that it may be very useful for target tracking and has naturally been incorporated into the geometric observer framework described above. In particular, given the values of a photometric variable, which is to be used for classifying the image pixels, the active contours are designed to converge to the shape that results in *maximal* discrepancy between the empirical distributions of the photometric variable inside and outside of the contours. This discrepancy is measured by means of the Bhattacharyya distance that proves to be an extremely useful tool for solving the problem at hand. The proposed methodology can be viewed as a generalization of the segmentation methods, in which the active contours maximize the difference between a finite number of empirical moments of the "inside" and "outside" distributions.

2.3.1 Bhattacharyya Flow

Bhattacharyya Distance

In order to facilitate the discussion, we just consider the case of two classes (i.e., the problem of segmenting an object of interest from the background), followed by describing the extension of the methodology to multi-object scenarios.

In the two class case, the segmentation problem is reduced to the problem of partitioning the domain of definition $\Omega \subset \mathbb{R}^2$ of an image $I(z)$ (with $z \in \Omega$) into two mutually exclusive and complementary subsets Ω_- and Ω_+. These subsets can be represented by their respective characteristic functions χ_- and χ_+, which can in turn be defined by means of a level set function $\Psi(z) : \Omega \to \mathbb{R}$ as $\chi_-(z) := H(-\Psi(z))$, $\chi_+(z) := H(\Psi(z))$ with $z \in \Omega$, where H denotes the Heaviside function.

As above, given a level set function $\Psi(z)$, its zero level set $\{z \mid \Psi(z) = 0, z \in \Omega\}$ is used to implicitly represent a curve. For the sake of concreteness, we associate the subset Ω_- with the support of the object of interest, while Ω_+ is associated with the support of corresponding background. In this case, the objective of active contour based image segmentation is given an initialization $\Psi_0(z)$, construct a *convergent* sequence of level set functions $\{\Psi_t(z)\}_{t>0}$ (with $\Psi_t(z)|_{t=0} = \Psi_0(z)$) such that the zero level set of $\Psi_T(z)$ coincides with the boundary of the object of interest for some $T > 0$.

We construct the sequence of level set functions via a gradient flow that minimizes a properly defined cost functional. In our approach, the latter is derived in the following manner. First, the image to be segmented $I(z)$ is transformed into a *vector-valued* image of its local features $\mathbf{J}(z)$. Note that the feature image $\mathbf{J}(z)$ ascribes to every pixel of $I(z)$ a N-tuple of associated features, and hence it can be formally represented as a map from Ω to $\mathbb{R}^N$. Subsequently, given a level set function $\Psi(z)$, the following two quantities are computed:

$$P_-(\boldsymbol{x} \mid \Psi(z)) = \frac{\int_\Omega K_-(\boldsymbol{x} - \mathbf{J}(z))\, \chi_-(z)\, dz}{\int_\Omega \chi_-(z)\, dz} = \frac{\int_\Omega K_-(\boldsymbol{x} - \mathbf{J}(z))\, H(-\Psi(z))\, dz}{\int_\Omega H(-\Psi(z))\, dz}, \tag{17}$$

and

$$P_+(\boldsymbol{x} \mid \Psi(z)) = \frac{\int_\Omega K_+(\boldsymbol{x} - \mathbf{J}(z))\, \chi_+(z)\, dz}{\int_\Omega \chi_+(z)\, dz} = \frac{\int_\Omega K_+(\boldsymbol{x} - \mathbf{J}(z))\, H(\Psi(z))\, dz}{\int_\Omega H(\Psi(z))\, dz}, \tag{18}$$

where $\boldsymbol{x} \in \mathbb{R}^N$, and $K_-(\boldsymbol{x})$ and $K_+(\boldsymbol{x})$ are two scalar-valued functions having compact or effectively compact supports. Provided that the *kernels* $K_-(\boldsymbol{x})$ and $K_+(\boldsymbol{x})$ are normalized to have unit integrals, the functions $P_-(\boldsymbol{x} \mid \Psi(z))$ and $P_+(\boldsymbol{x} \mid \Psi(z))$ given by (17) and (18) are *kernel-based estimates* of the probability density functions pdf of the image features observed over the sub-domains Ω_- and Ω_+, respectively.

The core idea of the our method is quite intuitive and it is based on the assumption that, for a properly selected subset of image features, the "overlap" between the informational contents of the object and of the background has to be minimal. In other words, if one thinks of the active contour as a *discriminator* that separates the image pixels into two subsets, then the optimal contour should minimize the *mutual information* between these subsets. Note that for the case at hand, minimizing the mutual information is equivalent to maximizing the Kullback-Leibler divergence between the pdf's associated with the "inside" and "outside" subsets of pixels. For the reasons discussed below, however, instead of the divergence, we have developed a method to maximize the Bhattacharyya distance between the pdf's. (The *Bhattacharyya distance* is defined to be $-\log$ of the integral given in (20) below which defines the *Bhattacharyya coefficient*.) Specifically, the optimal active contour $\Psi^\star(z)$ is defined as:

$$\Psi^\star(z) = \arg \inf_{\Psi(z)} \{\tilde{B}(\Psi(z))\}, \tag{19}$$

where

$$\tilde{B}(\Psi(z)) = \int_{\boldsymbol{x} \in \mathbb{R}^N} \sqrt{P_-(\boldsymbol{x} \mid \Psi(z))\, P_+(\boldsymbol{x} \mid \Psi(z))}\, d\boldsymbol{x}, \tag{20}$$

with $P_-(\boldsymbol{x} \mid \Psi(z))$ and $P_+(\boldsymbol{x} \mid \Psi(z))$ being given by the equations (17) and (18), respectively.

Gradient Flow

In order to derive a scheme for minimizing (20), we need to compute its first variation. Accordingly, the first variation of $\tilde{B}(\Psi(z))$ (with respect to $\Psi(z)$) is given by:

$$\frac{\delta \tilde{B}(\Psi(z))}{\delta \Psi(z)} = \frac{1}{2} \int_{\boldsymbol{x} \in \mathbb{R}^N} \left(\frac{\partial P_-(\boldsymbol{x} \mid \Psi(z))}{\partial \Psi(z)} \sqrt{\frac{P_+(\boldsymbol{x} \mid \Psi(z))}{P_-(\boldsymbol{x} \mid \Psi(z))}} + \frac{\partial P_+(\boldsymbol{x} \mid \Psi(z))}{\partial \Psi(z)} \sqrt{\frac{P_-(\boldsymbol{x} \mid \Psi(z))}{P_+(\boldsymbol{x} \mid \Psi(z))}} \right) d\boldsymbol{x}. \tag{21}$$

Differentiating (17) and (18) with respect to $\Psi(z)$, one obtains:

$$\frac{\partial P_-(\boldsymbol{x} \mid \Psi(z))}{\partial \Psi(z)} = \delta(\Psi(z)) \left(\frac{P_-(\boldsymbol{x} \mid \Psi(z)) - K_-(\boldsymbol{x} - \mathbf{J}(z))}{A_-} \right), \tag{22}$$

and

$$\frac{\partial P_+(\boldsymbol{x} \mid \Psi(z))}{\partial \Psi(z)} = \delta(\Psi(z)) \left(\frac{K_+(\boldsymbol{x} - \mathbf{J}(z)) - P_+(\boldsymbol{x} \mid \Psi(z))}{A_+} \right), \tag{23}$$

where $\delta(\cdot)$ is the delta function, and A_- and A_+ are the areas of Ω_- and Ω_+ given by $\int_\Omega \chi_-(z)\, dz$ and $\int_\Omega \chi_+(z)\, dz$, respectively.

By substituting (22) and (23) in (21) and combining the corresponding terms, one can arrive at:

$$\frac{\delta \tilde{B}(\Psi(z))}{\delta \Psi(z)} = \delta(\Psi(z))\, V(z), \tag{24}$$

where

$$V(z) = \frac{1}{2}\tilde{B}(\Psi(z))(A_-^{-1} - A_+^{-1}) + \tag{25}$$

$$+\frac{1}{2}\int_{\boldsymbol{x}\in\mathbb{R}^N} K_+(\boldsymbol{x} - \mathbf{J}(z))\,\frac{1}{A_+}\sqrt{\frac{P_-(\boldsymbol{x}\,|\,\Psi(z))}{P_+(\boldsymbol{x}\,|\,\Psi(z))}}\,d\boldsymbol{x} - \frac{1}{2}\int_{\boldsymbol{x}\in\mathbb{R}^N} K_-(\boldsymbol{x} - \mathbf{J}(z))\,\frac{1}{A_-}\sqrt{\frac{P_+(\boldsymbol{x}\,|\,\Psi(z))}{P_-(\boldsymbol{x}\,|\,\Psi(z))}}\,d\boldsymbol{x}.$$

Assuming the same kernel $K(\boldsymbol{x})$ is used for computing the last two terms in (25), i.e. $K(\boldsymbol{x}) = K_-(\boldsymbol{x}) = K_+(\boldsymbol{x})$, the latter can be further simplified to the following form:

$$V(z) = \frac{1}{2}\tilde{B}(\Psi(z))(A_-^{-1} - A_+^{-1}) + \frac{1}{2}\int_{\boldsymbol{x}\in\mathbb{R}^N} K(\boldsymbol{x} - \mathbf{J}(z))\,L(\boldsymbol{x}\,|\,\Psi(z))\,d\boldsymbol{x}, \tag{26}$$

where

$$L(\boldsymbol{x}\,|\,\Psi(z)) = \frac{1}{A_+}\sqrt{\frac{P_-(\boldsymbol{x}\,|\,\Psi(z))}{P_+(\boldsymbol{x}\,|\,\Psi(z))}} - \frac{1}{A_-}\sqrt{\frac{P_+(\boldsymbol{x}\,|\,\Psi(z))}{P_-(\boldsymbol{x}\,|\,\Psi(z))}}. \tag{27}$$

Introducing an artificial *time* parameter t, the gradient flow of $\Psi(z)$ that minimizes (20) is given by:

$$\Psi_t(z) = -\frac{\delta\tilde{B}(\Psi(z))}{\delta\Psi(z)} = -\delta(\Psi(z))\,V(z), \tag{28}$$

where the subscript t denotes the corresponding partial derivative, and $V(z)$ is defined as given by either (25) or (26).

From the viewpoint of statistical estimation, the cost function (20) can be thought of as accounting for the fidelity of estimation of the optimal level set function to observed features of $I(z)$. However, this cost function does not take into consideration some plausible properties of the optimal solution, and, as a result, minimizing (20) alone could be too sensitive to measurement noises and/or errors in the data. In order to alleviate this sensitivity, one can attempt to filter out the spectral components of the solution which belong to the noise subspace. Such filtering can be conveniently implemented using the Bayesian estimation framework, which allows one to find the most likely solution given both the observed data and a reasonable assumption regarding the nature of optimal $\Psi(z)$. One can also regularize the solution via constraining the length of the active contour, in which case the optimal level set $\Psi^\star(z)$ is given by:

$$\Psi^\star(z) = \arg\inf_{\Psi(z)}\left\{\tilde{B}(\Psi(z)) + \alpha\int_\Omega \|\nabla H(\Psi(z))\|\,dz\right\}, \tag{29}$$

where ∇ denotes the operator of gradient, $\|\cdot\|$ is the Euclidean norm, and $\alpha > 0$ is a regularization constant, which controls the compromise between fidelity and stability. The gradient flow associated with minimizing the cost functional in (29) can be shown to be equal to:

$$\Psi_t(z) = \delta(\Psi(z))\,(\alpha\,\kappa - V(z))\,, \tag{30}$$

where κ is the curvature of the active contour. All the segmentation results reported in the present study have been obtained via numerically solving (30) with the value of α set to be equal to 1.

2.3.2 Relationship with Kullback-Leibler Divergence

There exists a classical way to access similarity between the informational contents of two segmentation classes by means of mutual information. In our case, using the mutual information is equivalent to using the Kullback-Laibler divergence, whose symmetrized version is defined as:

$$D = \frac{1}{2}\int_{\mathbb{R}^N} p_1(\boldsymbol{x})\log\frac{p_2(\boldsymbol{x})}{p_1(\boldsymbol{x})}d\boldsymbol{x} + \frac{1}{2}\int_{\mathbb{R}^N} p_2(\boldsymbol{x})\log\frac{p_1(\boldsymbol{x})}{p_2(\boldsymbol{x})}d\boldsymbol{x}, \tag{31}$$

with $p_1(\boldsymbol{x})$ and $p_2(\boldsymbol{x})$ being the pdf's associated with the segmentation classes. Note that the Kullback-Laibler divergence can be turned into the Bhattacharyya coefficient under the substitution of the square root function for the logarithm function in (31):

$$\tilde{D} = \frac{1}{2}\int_{\mathbb{R}^N} p_1(\boldsymbol{x})\sqrt{\frac{p_2(\boldsymbol{x})}{p_1(\boldsymbol{x})}}d\boldsymbol{x} + \frac{1}{2}\int_{\mathbb{R}^N} p_2(\boldsymbol{x})\sqrt{\frac{p_1(\boldsymbol{x})}{p_2(\boldsymbol{x})}}d\boldsymbol{x} = \int_{\mathbb{R}^N}\sqrt{p_1(\boldsymbol{x})p_2(\boldsymbol{x})}d\boldsymbol{x} = B. \tag{32}$$

The above similarity between the two criteria raises the question of the advantages and disadvantages of using one criterion with respect to the other. Obviously, one of the advantages of using the Bhattacharyya coefficient is related to its analytical form, which is much simpler than that of the Kullback-Leibler divergence. As a result, the analytical derivations related to the former are usually much neater, shorter, and easier to interpret as compared to the latter.

Yet another, much more crucial difference between the criteria in (31) and (32) stems from the properties of the functions they employ. Specifically, the logarithm has an extremely high sensitivity to the variations of its argument for relatively small values of the latter (not to mention that it is undefined at zero). In the case of non-parametric density estimation, the above property of the logarithm is obviously a disadvantage, as it makes the Kullback-Leibler divergence prone to the errors caused by inaccuracies in estimating the *tails* of probability densities. On the other hand, the square root is a well-defined function in the vicinity of zero. Moreover, for relatively small values of its argument, the variability of the square root is considerably smaller than that of the logarithm. As a result, the Bhattacharyya coefficient is less susceptible to the influence of inaccuracies mentioned above.

2.3.3 Examples of Discriminative Features

In this subsection, a number of possible definitions of the feature vector are discussed. The set of examples given below is by no means complete, but it rather represents the features frequently used in practice.

Formally, the transition from the data image $I(z) : \Omega \to \mathbb{R}$ to the vector-valued image $\mathbf{J}(z) : \Omega \to \mathbb{R}^N$ of its features can be described by a transformation $\mathcal{W}\{\cdot\}$ applied to $I(z)$, i.e., $\mathbf{J}(z) = \mathcal{W}\{I(z)\}$. Hence, the question of selecting a useful set of image features is essentially equivalent to the question of defining a proper transformation $\mathcal{W}$. Perhaps, the simplest choice here is to define the feature image $\mathbf{J}(z)$ to be identical to $I(z)$, which corresponds to the case of $\mathcal{W}$ being identity and $N = 1$. In this case, the features used for the classification are the gray-levels of $I(z)$, and the resulting segmentation procedure is essentially *histogram based* [14].

Although the above choice has proven useful in numerous practical settings, it is definitely not the best possible for situations when both object and background have similar intensity patterns. In this case, it seems reasonable to take advantage of a relative displacement of these patterns with respect to each other via transforming $I(z)$ to the space of its partial derivatives. This transformation is performed by setting $\mathcal{W} \equiv \nabla$, in which case the feature space becomes two-dimensional, i.e., $N = 2$.

As a next logical step, one can smooth the above gradient $\nabla I(z)$ using a set of low-pass filters with progressively decreasing bandwidths. This construction brings us directly to the possibility to define $\mathcal{W}$ to be a *wavelet transform* [32]. Note that, in this case, each pixel of the resulting $\mathbf{J}(z)$ carries information on *multiresolution* (partial) derivatives of $I(z)$. It should be noted that using the wavelet coefficients as discriminative features for image segmentation has long been successfully used in numerous applications [58].

The dependency structure between the partial derivatives of $I(z)$ can be captured by the *structural tensor* defined as:

$$\mathbf{T}(z) = \begin{bmatrix} (\partial_{z_1} I(z))^2 & \partial_{z_1} I(z)\partial_{z_2} I(z) \\ \partial_{z_1} I(z)\partial_{z_2} I(z) & (\partial_{z_2} I(z))^2 \end{bmatrix}, \tag{33}$$

with $z = (z_1,\, z_2)$, and ∂_{z_1} and ∂_{z_2} denoting the corresponding partial derivatives. In this case, the feature space is three dimensional, as for each z, $\mathbf{J}(z)$ is defined to be equal to

$$\left[(\partial_{z_1} I(z))^2,\ \partial_{z_1} I(z)\partial_{z_2} I(z),\ (\partial_{z_2} I(z))^2\right].$$

Note, however, that this choice of the feature space is not linear since the structural tensor (33) is positive definite, and therefore the pixels of $\mathbf{J}(z)$ lie on a non-linear manifold. Fortunately, the availability of kernel-based methods for estimating probability densities defined over non-linear manifolds makes it possible to apply our approach in this situation as well.

Another interesting choice of $\mathbf{J}(z)$ can be followed in the scenarios, in which $I(z)$ appears as an element of a sequence of tracking images. In this case, $\mathbf{J}(z) : \Omega \to \mathbb{R}^2$ can be defined to be the *vector field of local displacements* of the gray-levels of $I(z)$. Specifically, let $\partial_t I(z)$ denotes the temporal (partial) derivative of $I(z)$. Let also $G(z)$ be a two component, column vector with its first and second components equal to $\partial_{z_1} I(z)\, \partial_t I(x)$ and $\partial_{z_2} I(z)\, \partial_t I(x)$, respectively. Then, the *gray-level constancy* constraint can be shown to result in the following least square solution for the local displacement field $\mathbf{v}(z)$:

$$\mathbf{v}(z) = -\mathbf{T}^{-1}(z)\, G(z), \quad \forall z \in \Omega, \tag{34}$$

with $\mathbf{T}(z)$ being the structure tensor given by (33). Consequently, in the case when the motion of the tracked object is independent of that of its background, the feature image can be defined by setting $\mathbf{J}(z) = \mathbf{v}(z)$. We note that this choice seems to be reasonable for many tracking scenarios, where the background motion is either negligible or associated with the ego-motion of camera, which rarely correlates with the dynamics of tracked objects.

The local moments of $I(z)$ [56], multiresolution versions thereof, and the local fractal dimension are among many other image features, which could be used for segmentation. Note that a combined use of all the features mentioned above is also possible. Thus, for example, by letting the feature vector $\boldsymbol{x} = (x_1, x_2, x_3)$ be composed of the intensity x_1 and local velocity components x_2, x_3, one can perform segmentation based on *both* gray-scale and motion information.

2.3.4 Kernel Based Estimation

The conditional densities $p(\boldsymbol{x} \,|\, \boldsymbol{x} \in \Omega_-)$ and $p(\boldsymbol{x} \,|\, \boldsymbol{x} \in \Omega_-)$ may be estimated using the kernel estimation method. In this subsection, we briefly consider the problem of defining the kernel's *bandwidth*, which should be consistent with the data size for the estimates to be reliable.

In the discrete setting, kernel density estimation amounts to approximating the (unknown) pdf $p(\boldsymbol{x})$ of an N-dimensional random vector $\boldsymbol{x}$ according to:

$$f(\boldsymbol{x}) = \frac{1}{n} \sum_{i=1}^{n} K_{\boldsymbol{\sigma}}(\boldsymbol{x} - \boldsymbol{x}^i), \tag{35}$$

where $\{\boldsymbol{x}^i\}_{i=1}^n$ are n independent realizations of $\boldsymbol{x}$. In (35), the kernel $K_{\boldsymbol{\sigma}}(\boldsymbol{x})$ is parameterized by a vector $\boldsymbol{\sigma}$ and has a unit integral, i.e. $\int_{\mathbb{R}^N} K_{\boldsymbol{\sigma}}(\boldsymbol{x})\, d\boldsymbol{x} = 1$.

A number of possible definitions of the kernel $K_{\boldsymbol{\sigma}}(\boldsymbol{x})$ are possible, among which the most popular to take $K_{\boldsymbol{\sigma}}(\boldsymbol{x})$ to be a $Gaussian$ pdf, and this is the choice followed in this proposal. In order to facilitate the numerical implementation of the density estimation, we use a separable (isotropic) form of the Gaussian pdf, in which case $K_{\boldsymbol{\sigma}}(\boldsymbol{x})$ is defined as:

$$K_{\boldsymbol{\sigma}}(\boldsymbol{x}) = (2\pi)^{-N/2} \prod_{k=1}^{N} \sigma_k^{-1} \exp\left\{-x_k^2 / 2\sigma_k^2\right\}, \tag{36}$$

where $x_1, x_2, \ldots, x_N$ are the coordinates of $\boldsymbol{x}$, and the standard deviations $\boldsymbol{\sigma} = \{\sigma_k\}_{k=1}^N$ control the extension of $K_\sigma(\boldsymbol{x})$ along each of the independent directions. It should be noted that the separability of the kernel in (36) does not imply the separability of the estimate $f(\boldsymbol{x})$ that is now given by:

$$f(\boldsymbol{x}) = \frac{1}{n} \sum_{i=1}^{n} K_{\sigma_1}(x - x_1^i)\, K_{\sigma_2}(x - x_2^i) \cdots K_{\sigma_N}(x - x_N^i), \tag{37}$$

with $K_{\sigma_k} = (\sqrt{2\pi\sigma_k^2})^{-1} \exp\left\{-x_k^2/2\sigma_k^2\right\}$ for $k \in \{1, 2, \ldots, N\}$.

The kernel method of density estimation has proven valuable in numerous applications. It is well known, however, that effective use of this method requires proper choice of the *bandwidth* parameters σ_k. When insufficient smoothing is done (i.e., the bandwidth parameters are too small), the resulting density estimate is too rough and contains spurious features that are artifacts of the sampling process. On the other hand, when excessive smoothing is done, important features of the underlying structure are smoothed away. Consequently, to optimize the accuracy of the estimates in (17) and (18), the bandwidth parameters $\{\sigma_k\}_{k=1}^N$ should be properly defined. This will be a topic of investigation in our research, and in particular we are considering an expectation maximization based approach for this purpose.

2.3.5 Multi-Object Case

In order to extend the applicability of the proposed method to the case when images are composed of more than two segmentation classes, a multi-class version will now be described. In particular, in such a scenario, the image domain Ω is considered to be a union of M (mutually exclusive) subdomains $\{\Omega_k\}_{k=1}^M$, each of which is associated with a corresponding (conditional) pdf $p_k(\boldsymbol{x})$. Consequently, the first step to be done is to generalize the definition of the Bhattacharyya coefficient to the case of M densities. Such a natural generalization is known as the *average* Bhattacharyya coefficient as given by [22]:

$$B = \frac{2}{M(M-1)} \sum_{\substack{i,\, j=1 \\ j>i}}^{M} \int_{\mathbb{R}^N} \sqrt{p_i(\boldsymbol{x})\, p_j(\boldsymbol{x})}\, d\boldsymbol{x}. \tag{38}$$

The above coefficient represents a *cumulative* measure of discrepancy between all the possible pairs of the densities under consideration. Note that the normalization constant $2/(M(M-1))$ guarantees that the coefficient (38) takes its values in the interval between 0 and 1.

The additivity of the construction in (38) makes it trivial to derive the corresponding gradient flow. In fact, each additive term of the cost functional can be differentiated independently so that the resulting gradient flow is given as a *sum* of the gradient flows related to the additive components in (38). In order to complete the multi-class formulation of the Bhattacharyya flow, we will need to briefly describe how one would use level sets in this context.

Obviously, in the multi-class case, using only *one* level set function would be insufficient to solve the problem at hand. Consequently, we follow the multiphase segmentation formulation of [63] that uses p level set functions $\{\Psi_i(z)\}_{i=1}^p$, which are capable of segmenting the image $I(z)$ into (up to) 2^p regions. In particular, since one level set function partitions the image domain Ω into two sub-domains, p level set functions can partition Ω into 2^p sub-domains, each of which is labeled by the *signs* of the level set functions $\{\Psi_i(z)\}_{i=1}^p$ in that sub-domain. The formulae for the multiphase gradient flow corresponding to (38) can be easily obtained by plugging the results of Section 2.3.1 into the "templates" derived in [63], *mutatis mutandis*.

2.3.6 Information-Theoretic Segmentation and Blind Source Separation

We have pursued a number of directions for the use of segmentation in visual tracking. First of all, there is a similarity between the problem of image segmentation by means of active contours and the problem of *blind source separation* as a specific instance of *independent component analysis* (ICA) [20]. The latter is a reconstruction problem, in which a number of unknown source signals have to be recovered from measurements of their *algebraic* mixtures. This problem has inspired the proposal of numerous solutions, many of which are based on finding the directions - *independent components* - in a multi-dimensional space, along which the mixtures (or the projections thereof) are *as independent as possible*. In this connection, a number of information-based criteria, such as the Kullback-Leibler divergence, have been employed. Returning to the problem of segmentation, one can think of a data image as a *geometric* mixture of sources, i.e., of segmentation classes. In such a case, the active contour acts in a certain sense as an independent component when trying to separate the image into as many independent regions as possible. It is quite

interesting to note that the method proposed here employs the concept and tools, which are very much similar to those used in ICA. This has been a nice research direction in the just completed AFOSR program.

References

[1] R. Abraham, J. E. Marsden, and R. Ratiu, *Manifolds, tensor analysis, and applications: 2nd edition*, Springer-Verlag, New York, 1988.

[2] S. Angenent, S. Haker, and A. Tannenbaum, "Minimizing flows for the Monge–Kantorovich problem," *SIAM J. Math. Analysis*, vol. 35, pp. 61–97, 2003.

[3] S. Arulampalam, S. Maskell, N. J. Gordon, and T. Clapp, "A tutorial on particle filters for on-line non-linear/non-gaussian Bayesian tracking," *IEEE Transactions of Signal Processing*, vol. 50, pp. 174–188, February 2002.

[4] A. Bhattacharyya, "On a measure of divergence between two statistical populations defined by their probability distributions," *Bull. Calcutta Math. Soc.*, vol. 35, pp. 99–109, 1943.

[5] G. Ben-Arous, A. Tannenbaum, and A. Tannenbaum, "Stochastic approximations of curvature flows," *Journal of Differential Equations*, vol. 195, pp. 119-142, 2003.

[6] A. Birnbaum, "On the foundations of statistical inference (with discussion)," *J. Amer. Statist. Assoc.*, vol. 57, pp. 269–326, 1962.

[7] A. Blake and M. Isard, *Active Contours*, Springer-Verlag, 1998.

[8] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic snakes," *Int. J. Computer Vision*, 1997.

[9] T. Chan and L. Vese, "Active contours without edges," *IEEE Trans. Image Processing*, vol. 10, no. 2, pp. 266–277, 2001.

[10] A. Doucet, N. deFreitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*, Springer, 2001.

[11] J. C. Doyle, B. Francis, and A. Tannenbaum, *Feedback Control Theory*, McMillan, New York, 1991.

[12] D. Freedman and T. Zhang, "Active contours for tracking distributions," *IEEE Trans. Image Processing*, vol. 13, no. 4, pp. 518–526, Apr. 2004.

[13] D. Fry, *Shape Recognition Using Metrics on the Space of Shapes*, Ph.D. Thesis, Harvard University, 1993.

[14] C. A. Glasbey, "An analysis of histogram-based thresholding algorithms," *Graphical Models and Image Processing*, vol. 55, pp. 532–537, 1993.

[15] N.J. Gordon, D.J. Salmond, and A.F.M. Smith, "Novel approach to nonlinear/nongaussian bayesian state estimation," *IEE Proceedings-F (Radar and Signal Processing)*, pp. 140(2):107–113, 1993.

[16] S. Haker, G. Sapiro, and A. Tannenbaum, "Knowledge-based segmentation of SAR data with learned priors," *IEEE Trans. Image Processing*, vol. 9, pp. 298–302, 2000.

[17] S. Haker, L. Zhu, S. Angenent, and A. Tannenbaum, "Optimal mass transport for registration and warping" *Int. Journal Computer Vision*, vol. 60, pp. 225–240, 2004.

[18] J. P. Holman, *Heat Transfer*, McGraw-Hill, New York, 1981.

[19] B. K. P. Horn, *Robot Vision,* MIT Press, Cambridge, Mass., 1986.

[20] A. Hyvarinen, J. Karhunen, and E. Oja, *Independent Component Analysis*, John Wiley and Sons, New York, 2001.

[21] M. Isard and A. Blake, "Condensation – conditional density propagation for visual tracking," *International Journal of Computer Vision*, vol. 29, no. 1, pp. 5–28, 1998.

[22] T. Kailath, "The divergence and Bhattacharyya distance measures in signal selection," *IEEE Trans. Commun. Technol.*, vol. COM-15, no. 1, pp. 52–60, Feb. 1967.

[23] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: active contour models," *International Journal of Computer Vision*, pp. 321-331, 1988.

[24] C. Y. Kao, S. Osher, and Y.-H. Tsai, "Fast sweeping methods for static Hamilton-Jacobi equations," UCLA, Tech. Rep. 03-75, 2003.

[25] M. Katsoulakis and A. T. Kho, "Stochastic curvature flows: Asymptotic derivation, level set formulation and numerical experiments," *Journal of Interfaces and Free Boundaries*, vol. 3, pp. 265–290, 2001.

[26] S. Kichenassamy, A. Kumar, P. Olver, A. Tannenbaum, and A. Yezzi, "Conformal curvature flows: from phase transitions to active vision," *Archive for Rational Mechanics and Analysis*, vol. 134, pp. 275–301, 1996.

[27] S. Kullback and R. A. Leibler, "On information and sufficiency," *Ann. Math. Stat.*, vol. 22, pp. 79–86, 1951.

[28] S. Kullback, *Information Theory and Statistics*, Wiley, New York, 1959.

[29] S. Lankton and A. Tannenbaum, "Localizing region-based active contours," *IEEE Trans. Image Processing* vol. 17, pp. 2029–2039, 2008.

[30] E. L. Lehmann, *Testing Statistical Hypotheses*, John Wiley and Sons, New York, 1986.

[31] D. G. Luenberger, "An introduction to observers," *IEEE Transactions on Automatic Control*, vol. 16, no. 6, pp. 596–602, 1971.

[32] S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, pp. 674–693, 1989.

[33] O. Michailovich, Y. Rathi, and A. Tannenbaum, "Image segmentation using active contours driven by the Bhattacharyya gradient flow," *IEEE Trans. Image Processing*, vol. 16, pp. 2787–2801, 2008.

[34] D. Mumford and J. Shah, "Optimal approximation by piecewise smooth functions and associated variational problems," *Commun. Pure Applied Mathematics*, vol. 42, pp. 577–685, 1989.

[35] M. Niethammer, P. Vela, and A. Tannenbaum, "Geometric observers for dynamically evolving curves," *IEEE PAMI*, vol. 30, pp. 1093-1108, 2008.

[36] M. Niethammer, A. Tannenbaum, and P. Vela, "On the evolution of closed curves by means of vector distance functions" *Int. Journal Computer Vision*, vol. 65, pp. 5–27, 2005.

[37] M. Niethammer, A. Tannenbaum, and S. Angenent, "Dynamic geodesic snakes for visual tracking," *IEEE Transactions on Automatic Control*, vol. 51, pp. 562–579, 2006.

[38] S. J. Osher and J. A. Sethian, "Fronts propagation with curvature dependent speed: Algorithms based on hamilton-jacobi formulations," *Journal of Computational Physics*, vol. 79, pp. 12–49, 1988.

[39] S. Osher and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*, Springer Verlag, 2003.

[40] N. Paragios and R. Deriche, "Geodesic active regions: a new framework to deal with frame partition problems in computer vision," *Journal of Visual Communication and Image Representation*, vol. 13, pp. 249–268, 2002.

[41] N. Paragios and R. Deriche, "Geodesic active contours and level sets for the detection and tracking of moving objects," *IEEE Trans. on Pattern analysis and Machine Intelligence*, vol. 22, pp. 266–280, 2000.

[42] N. Peterfreund, "Robust tracking of position and velocity with Kalman snakes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, pp. 564–569, 1999.

[43] E. Pichon, A. Tannenbaum, and R.Kikinis, "Statistically based flow for image segmentation," *Medical Image Analysis*, vol. 8, pp. 267–274, 2004.

[44] Y. Rathi, N. Vaswani, A. Tannenbaum, and A. Yezzi, "Tracking moving and deforming shapes using a particle filter," *Proceedings of CVPR*, 2005.

[45] Y. Rathi, N. Vaswani, and A. Tannenbaum, "A generic framework for tracking using particle filter with dynamic shape prior," *IEEE Trans. Image Processing*, vol. 16, pp. 1370–1382, 2007.

[46] Y. Rathi, N. Vaswani, A. Yezzi, and A. Tannenbaum, "Tracking deforming objects using particle filtering for geometric active contours" *IEEE PAMI*, vol. 29, pp. 1470–1475, 2007.

[47] N. Vaswani, Y. Rathi, A. Yezzi, and A. Tannenbaum, "Deform PF-MT (Particle Filter with Mode Tracker) for tracking nonaffine contour deformations," *IEEE Trans. Image Processing*, vol. 19, pp. 841–857, 2010.

[48] B. Ristic, S. Arulampalam, and N. Gordon, "Beyond the Kalman filter: particle filters for tracking applications," Wiley, New York, 2004.

[49] G. Sapiro, *Geometric Partial Differential Equations and Image Analysis* Cambridge University Press, 2001.

[50] G. Sapiro and A. Tannenbaum, "Invariant curve evolution and image analysis," *Indiana University J. of Mathematics*, vol. 42, pp. 985–1009, 1993.

[51] R. Sedgewick, *Algorithms*, Addison-Wesley, 1992.

[52] S. Soatto D. Cremers, S. Osher, "Kernel density estimation and intrinsic alignment for knowledge-driven segmentation: Teaching level sets to walk," *Pattern Recognition,Tbingen, Springer LNCS*, vol. 3175, no. 3, pp. 36–44, 2004.

[53] A. Tannenbaum, "Three snippets of curve evolution in computer vision," *Mathematical and Computer Modelling Journal*, vol. 24, pp. 103–119, 1996.

[54] H. Tao, H. Sawhney, and R. Kumar, "A sampling algorithm for tracking multiple objects," in *Proceedings of Vision Algorithms, ICCV*, 1999.

[55] D. Terzopoulos and R. Szeliski, "Tracking with Kalman snakes," in *Active Vision* edited by A. Blake and A. Yuille, MIT Press, pp. 3–20, 1992.

[56] M. Tuceryan, "Moment based texture segmentation," *Pattern Recognition Letters*, vol. 15, pp. 659–668, July 1994.

[57] D. Tweed and A. Calway, "Tracking many objects using subordinated condensation," *The British Machine Vision Conference*, pp. 283–292, 2002.

[58] M. Unser, "Texture classification and segmentation using wavelet frames," *IEEE Trans. Image Processing*, vol. 11, no. 4, pp. 1549–1560, Apr. 1995.

[59] R. van der Merwe, N. de Freitas, A. Doucet, and E. Wan, "The unscented particle filter," in *Advances in Neural Information Processing Systems 13*, Nov. 2001.

[60] N. Vaswani, Y. Rathi, A. Yezzi, and A. Tannenbaum, "Particle filtering for continuous closed curves," *IEEE CDC*, 2005.

[61] N. Vaswani, *Change Detection in Stochastic Shape Dynamical Models with Applications in Activity Modeling and Abnormality Detection*, Ph.D. Thesis, ECE Dept, University of Maryland at College Park, August 2004.

[62] P. Vela, M. Niethammer, G. Pryor, A. Tannenbaum, R. Butts, and D. Washburn, "Knowledge-based segmentation for tracking through deep turbulence," to appear in *IEEE Trans. Control Technology*, 2008.

[63] L. A. Vese and T. F. Chan, "A multiphase level set framework for image segmentation using the Mumford and Shah model," *International Journal of Computer Vision*, vol. 50, no. 3, pp. 271–293, 2002.

[64] L. Xu and M. I. Jordan, "On convergence properties of the EM algorithm for Gaussian mixtures," *Neural Computation*, vol. 8, no. 1, pp. 129–151, Jan. 1996.

[65] A. Yezzi and S. Soatto, "Deformotion: Deforming motion, shape average and the joint registration and approximation of structures in images," *Internaitonal Journal of Computer Vision*, vol. 53, pp. 153–167, 2003.

[66] T. Zhang and D. Freedman, "Tracking objects using density matching and shape priors," *Proceedings of IEEE International Conference on Computer Vision*, pp. 1950–1954, 2003.